

Fig. 3 Block diagram of RBFNN based estimator

DE algorithm involves three basic operations viz., mutation, recombination, and selection. The step-wise procedure for the development of RBFDE is described below.

Step 1 Randomly initialize $i = 1, 2, 3, \dots, NP$ number of target or population vectors, $\vec{x}_{i,G}$ between 0 to 1, where each i th individual of the population vector represents parameters of the RBFDE model. The i th target vector of G th generation, $\vec{x}_{i,G}$ is given as $\vec{x}_{i,G} = \{w_i, c_i, \sigma_i\}$.

Step 2 Repeat step 3 with each target vector $\vec{x}_{i,G}$ for $i = 1, 2, 3, \dots, NP$.

Step 3 a) Give K numbers of input patterns to the RBF network sequentially with each pattern having dimension n .

b) For each one of the K input patterns, obtain corresponding network output using i th target vector $\vec{x}_{i,G}$ as the parameters of the network and compare it with the corresponding desired output to get an error using (3). For K patterns, the K number of error values will be obtained.

c) Calculate $f(\vec{x}_{i,G})$ using (7), where $f(\vec{x}_{i,G})$ represents the fitness function i.e. mean square error (MSE).

$$f(\vec{x}_{i,G}) = \frac{\sum_{j=1}^K e^2}{K} \tag{7}$$

Step 4 Obtain $f_{min}(\vec{x}_{i,G})$ and represent the corresponding $\vec{x}_{i,G}$ as the $\vec{x}_{best,G}$ for G th generation.

Step 5 Choose a scaling factor $F \in [0, 1]$ and a cross over ratio $CR, \in [0, 1]$ and repeat step 6 to step 15 until the desired minimum MSE is obtained.

Step 6 Repeat the steps from 7 to 8 for $i = 1, 2, 3, \dots, NP$ times

Step 7 Randomly choose two indices r_1, r_2 from 1 to NP , such that, $r_1 \neq r_2 \neq i$.

Step 8 Compute the mutant vector $\vec{v}_{i,G}$ for each target vector $\vec{x}_{i,G}$ for G th generation as

$$\vec{v}_{i,G} = \vec{x}_{i,G} + F * (\vec{x}_{r_1,G} - \vec{x}_{r_2,G}) + F * (\vec{x}_{best,G} - \vec{x}_{i,G}). \tag{8}$$

Step 9 Repeat the steps from 10 to 11 for $i = 1, 2, 3, \dots, NP$ times

Step 10 Randomly choose an index r_3 between 1 to d and repeat step 11 for $j=1$ to d , where d is the dimension of the target or population vector.

Step 11 Generate a random number $rand \in [0, 1]$ and compute the trial vector $\vec{U}_{j,i,G}$ by recombination operation, which replaces the previously successful individuals with mutant vector as

$$\vec{U}_{j,i,G} = \begin{cases} \vec{v}_{j,i,G} & \text{if } (rand \leq CR) \text{ or } j = r_3 \\ \text{else} \\ \vec{x}_{j,i,G} \end{cases} \tag{9}$$

Step 12 For each trial vector $\vec{U}_{j,i,G}, i = 1, 2, 3, \dots, NP$, evaluate $f(U_{i,G})$, which is a mean square error (MSE). (Similar to step 3)

Step 13 Repeat the step14 for $i = 1, 2, 3, \dots, NP$

Step 14 Finally, the next generation of NP number of target/population vector $\vec{x}_{i,G+1}$ is selected based on survival of the fittest criteria as

$$\vec{x}_{i,G+1} = \begin{cases} \vec{U}_{i,G} & \text{if } f(\vec{U}_{i,G}) \leq f(\vec{x}_{i,G}) \\ \text{else} & \\ \vec{x}_{i,G} & \end{cases} \quad (10)$$

Step 15 Obtain $f_{\min}(\vec{x}_{i,G+1})$ and represent it as $\vec{x}_{best,G+1}$ for the next generation.

Step 16 Stop

Pictorial representation of the DE algorithm is shown in Fig. 4.

2.2.3 Particle swarm optimization based RBF neural network estimator

In this approach parameters of the RBFNN model i.e. $\{w_{ij}, c_{ij}, \sigma_j\}$, as described in Sect. 2.2.1, are updated using the PSO algorithm. The PSO [51–53] is a metaheuristics optimization algorithm inspired by the paradigm of swarm intelligence which mimics the social behavior of animals like fish and birds. It is successfully applied to various applications in engineering and science [54–56]. The algorithm uses a fixed number of particles that represent the parameters of RBFNN. Each particle updates its current velocity and position by its own experience called personal best (*p-best*) and by the social experience of the swarm called global best (*g-best*). Steps involved in PSO are briefly described below:

Step 1 Initialize fix number of particles with random position and velocity uniformly distributed over the search space.

Step 2 Evaluate the fitness of each particle according to the objective function

Step 3 Record *pbest* for each particle and *g-best* of the swarm.

Step 4 Update velocity of each particle

Step 5 Update the position of each particle.

Step 6 Update *pbest* and *gbest*

Step 7 Repeat the steps from 2 to 6 until the termination condition is satisfied and stop.

Pictorial representation of the PSO algorithm is shown in Fig. 5.

2.2.4 Multi-layer artificial neural network (MLANN)

MLANN, suggested by Haykin [57] is successfully employed in many applications to solve the regression problem. MLANN architecture considered for this proposed investigation consists of an N-5-1 structure. N represents the number of input features. Optimum results are obtained with 5 neurons in the intermittent hidden layer. Desired ET_0 estimates are obtained at output neurons. Hyperbolic tangent (tanh) is used as an activation function in every processing neuron. The training of the network is done by a conventional back-propagation algorithm which is based on the error-correcting learning rule to update the weights and bias of each neuron in different layers.

2.3 Empirical models

Weekly ET_0 for the study locations is also computed using empirical methods of FAO56-PM, Blaney-Criddle, Open Pan, Turc, and Hargreaves from available meteorological data. A brief description regarding empirical approaches considered in this investigation and the corresponding input meteorological parameter requirement are listed in Table 2. The description regarding different climate based empirical methods considered in this investigation is not included in this paper. More details regarding these empirical approaches can be obtained from basic references [1, 5–7].

2.4 Performance evaluation measures

Comparative analysis of estimated ET_0 obtained with different soft computing models and empirical methods considered for the investigation is carried out by computing performance evaluation measures, namely, mean square percentage error (MAPE), root mean square error (RMSE), determination coefficient (R^2) and efficiency factor (EF) proposed by Nash and Sutcliffe (NSE) [58]. The mathematical expression of different evaluation measures is as follows.

$$MAPE = \frac{1}{n} \sum_{i=1}^n \left| \frac{(Out_{obs} - Out_{est})}{Out_{obs}} \right| \times 100 \quad (11)$$

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (Out_{est} - Out_{obs})^2} \quad (12)$$

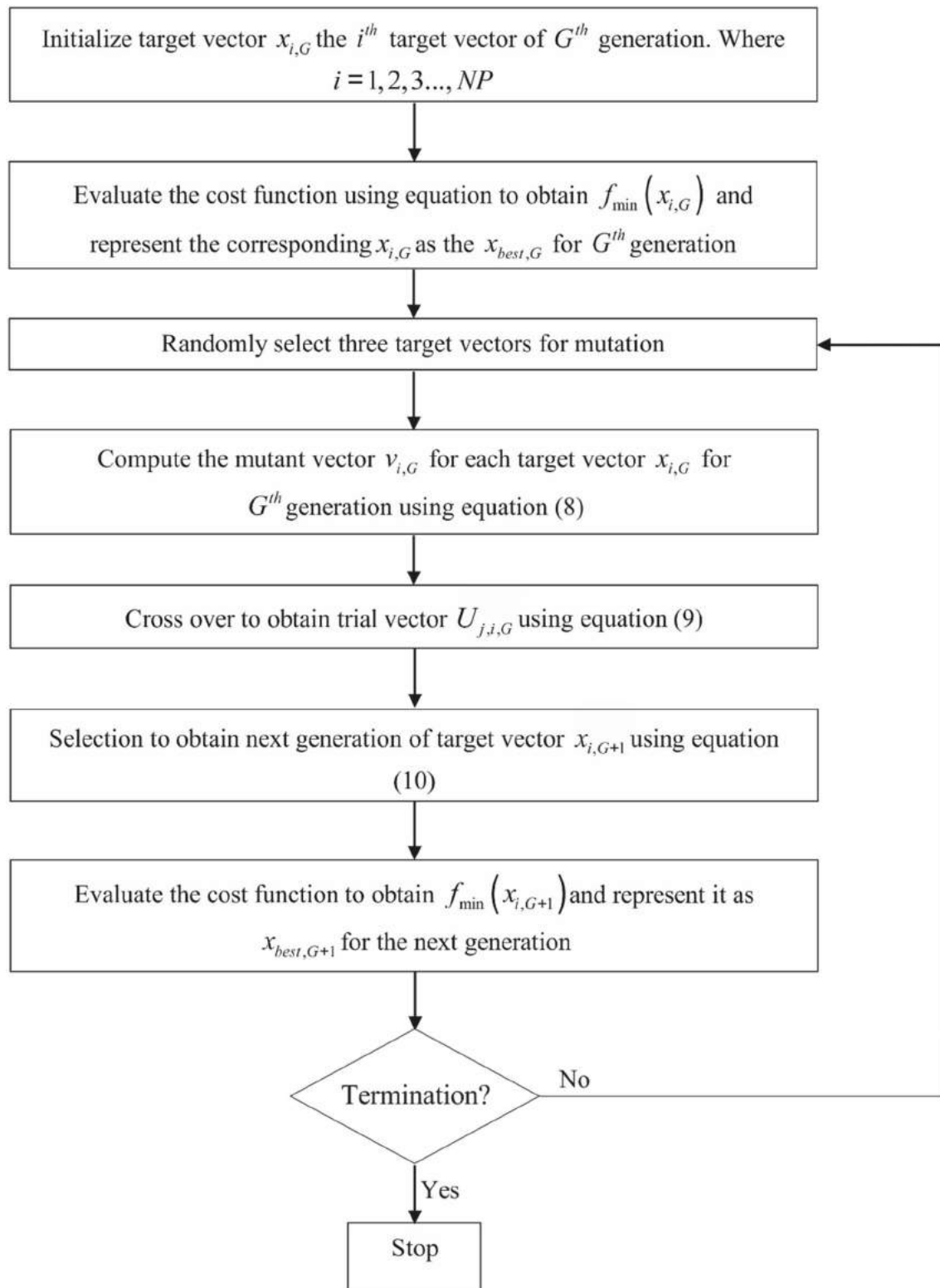


Fig. 4 Flowchart differential evolution algorithm

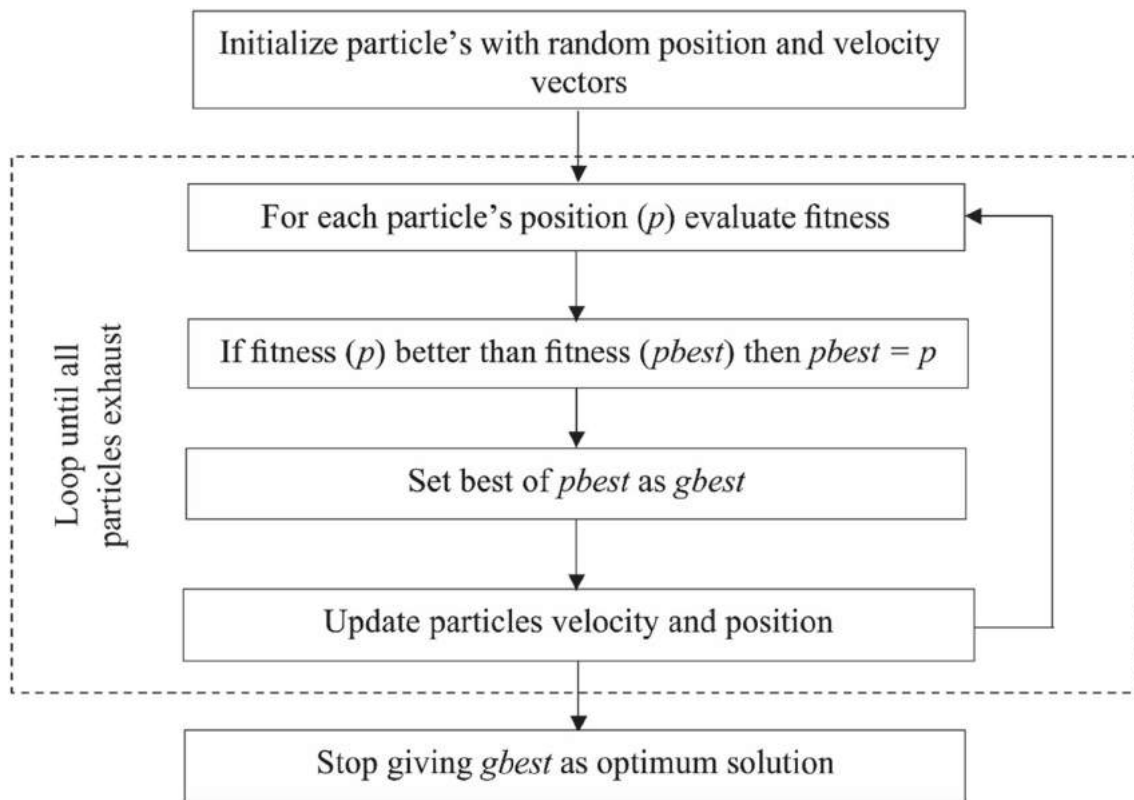


Fig. 5 Flowchart particle swarm optimization algorithm

Table 2 Details of empirical models used to compute ET₀

Empirical methods	Input combination	Empirical equation
Hargreaves [5]	$T_{max} T_{min}$	$ET_0 = 0.0023 \times R_a \times T_d \times 0.5 \times (T_m + 17.8)$
Turc [6]	$T_{max} T_{min} BSS$	$ET_0 = \frac{0.40 \times T_m (R_s + 50)}{(T + 15)}$
Blaney-Criddle [7]	$T_{max} T_{min} RH_1 \& RH_2, WS, BSS$	$ET_0 = a + b \times p(0.46 \times T_m + 8.13)$
Open Pan [7]	$WS, EP, RH_1 \& RH_2$	$ET_0 = k_p \times EP$
FAO56-PM [1]	$T_{max} T_{min} RH_1 \& RH_2, WS, BSS$	$ET_0 = \frac{0.408 \Delta (R_n - G) + \gamma \frac{900}{T + 273} u_2 (e_s - e_a)}{\Delta + \gamma (1 + 0.34 u_2)}$

$$R^2 = \frac{\left(\sum_{i=1}^n (Out_{obs} - \overline{Out_{obs}}) (Out_{est} - \overline{Out_{est}}) \right)^2}{\sum_{i=1}^n (Out_{obs} - \overline{Out_{obs}})^2 \sum_{i=1}^n (Out_{est} - \overline{Out_{est}})^2} \tag{13}$$

$$EF = NSE = 1 - \frac{\sum_{i=1}^n (Out_{est} - Out_{obs})^2}{\sum_{i=1}^n (Out_{obs} - \overline{Out_{obs}})^2} \quad (-\infty \leq EF \leq 1) \tag{14}$$

where Out_{obs} denotes the target and Out_{est} denotes model estimated ET₀ values. n is the number of testing patterns. Low MAPE and RMSE values represent the close agreement between desired and estimated output. Similarly,

R² and EF values close to 1 are also indicators of a higher accuracy level of the model.

3 Results and discussion

The key objective of this investigation is to examine the potential of different evolutionary optimized hybrid (RBFDE, RBFPSO) and conventional (RBFNN, MLANN) soft computing approaches with available climatic features for estimation of ET₀ comparable to FAO56-PM ET₀. Input features combination of different models is decided based on empirical approaches of Hargreaves, Turc, Open Pan, Blaney-Criddle, and FAO56-PM ET₀ listed in the previous section. These soft computing models are categorized

into type I to type V models. Like the Hargreaves method, type I models include only T_{max} and T_{min} as input features, whereas Type II soft computing models include BSS with temperature, which is equivalent to the Turc approach. In type III soft computing models, EP, RH_1 , RH_2 , and WS are considered as input features similar to that of the Open Pan empirical approach. Type IV models include six weather parameters (T_{max} , T_{min} , BSS, WS, and RH_1 and RH_2) equivalent to the Blaney-Criddle empirical approach. Another category of soft computing model termed Type V models are developed using T_{max} , T_{min} , BSS, and WS since these weather parameters exhibit positive correlations with ET_0 . The input feature EP, which is also positively correlated with ET_0 , is not included in type V soft computing models as obtaining EP data is very difficult. Input feature combinations used in different types of soft computing models and their equivalent empirical models are shown in Table 3. Weekly meteorological data of Raipur, Jagdalpur, and Ambikapur from 2001 to 2015 (80%) are used for model calibration or training, whereas the recent 4 years (20%) of the weekly meteorological data from 2016 to 2019 are used for model validation.

Soft computing models RBFDE, RBFPSO, RBFNN, and MLANN are coded in MATLAB as per the design and the

learning algorithm described earlier. Simulation studies are carried out with a different input features combination to test the sensitivity of the soft computing approach to control parameters until a satisfactory accuracy level is achieved for estimation of $FAO56-ET_0$ for different study locations. Detailed information regarding modeling strategies and respective control parameters that produce optimum results during the simulation process are shown in Table 4 for different soft computing models.

Calibration of RBFDE, RBFPSO, RBFNN, and MLANN models is done using the above-listed network parameters with training datasets of all the three study locations, Raipur, Jagdalpur, and Ambikapur. During the training process, input patterns are given to the model sequentially and the corresponding estimated output is obtained at the output layer after completion of the forward pass (Fig. 3). The estimated output is compared with the corresponding target $FAO56-ET_0$ output to compute the instantaneous error cost function. Real-time update of the model parameters is done in each instance to minimize the squared error using respective evolutionary (DE and PSO) and conventional back-propagation learning algorithms (RBFNN and MLANN). The process continues until all the available training input patterns for model calibration gets

Table 3 Input feature combinations used in soft computing and empirical models

Type	Soft computing models				Equivalent empirical model	Input feature combinations
	RBFDE	RBFPSO	RBFNN	MLANN		
I	RBFDE1	RBFPSO1	RBFNN1	MLANN1	Hargreaves	T_{max} , T_{min}
II	RBFDE2	RBFPSO2	RBFNN2	MLANN2	Turc	T_{max} , T_{min} , BSS
III	RBFDE3	RBFPSO3	RBFNN3	MLANN3	Open Pan	EP, RH_1 & RH_2 , WS
IV	RBFDE4	RBFPSO4	RBFNN4	MLANN4	Blaney-Criddle	T_{max} , T_{min} , RH_1 & RH_2 , WS, BSS
V	RBFDE5	RBFPSO5	RBFNN5	MLANN5	–	T_{max} , T_{min} , BSS, WS

Table 4 Parameters of the soft computing models used for simulation

Model parameters	RBFDE	RBFPSO	RBFNN	MLANN
Dimension of input features (<i>D</i>)	2, 3, 4 & 6	2,3,4 & 6	2, 3 4 & 6	2,3,4 & 6
Normalization of input features	0 to 1	0 to 1	0 to 1	– 1 to 1
Normalization method	$\frac{x_k - x_{min}}{x_{min} - x_{max}}$, where x_k represents <i>k</i> th instance			
Initial values of the model parameters	0 to 1	0 to 1	0 to 1	– 1 to 1
Number of centers for RBF based models (<i>c</i>)	10	10	10	–
Number of neurons in intermediate layer	–	–	–	5
Output neuron	1	1	1	1
Activation function	Gaussian	Gaussian	Gaussian	<i>tanh</i>
Learning algorithm	DE	PSO	RBF update rules	Back-propagation
Convergence coefficients/control parameters/	F=0.9 CR=0.9	Constriction coefficient $C_1 = C_2 = 1.49$ Inertia weight <i>w</i> Linearly decreased from 0.9 to 0.4	$\eta_1 = \eta_2 = \eta_3 = 0.01$	$\eta = 0.01$
Number of population/particles	$5 \times d$	50	–	–

exhausted. This completes one cycle called an epoch. At the end of each epoch, the mean square error is computed and stored for each epoch to examine the learning characteristic of soft computing models. The iterative process is repeated several times until MSE is minimized to a desired low value nearly close to zero. This completes the supervised learning process and model parameters are then fixed to constitute soft computing models. A similar calibration process is adopted for all soft computing approaches.

To test the performance of different soft computing models, test data patterns are then presented sequentially at the input layer of the model and through forward pass respective estimated ET_0 is obtained at the output layer for all the test patterns. These ET_0 estimates are then compared with corresponding target FAO56-PM ET_0 values. Performance evaluation measures, MAPE (%), RMSE ($mm\ week^{-1}$), R^2 and NSE as described in the previous section are then computed using desired and estimated output of different types of soft computing models and equivalent empirical approaches for comparison of model performance, which ultimately leads to model selection. The computed values of performance evaluation measures for different types of soft computing models and

equivalent empirical approaches considered are listed in Tables 5 and 6 for all three locations. Comparative results of the analysis are discussed below:

- i. For type I soft computing models, MAPE ranges from lowest of 7.4 for RBFDE1 and RBFDE2 (at Raipur) to highest of 11.8 for MLANN1 (at Jagdalpur), whereas MAPE obtained with Hargreaves model is comparatively very high and ranges between 22.6 (at Raipur) to 30.3 (at Ambikapur).
- ii. Type II soft computing models produce improved ET_0 estimates with low MAPE compared to type I models. For type II models, MAPE ranges from the lowest of 4.9 with RBFDE2 (at Raipur) to a high of 10.2 with MLANN2 (at Raipur). MAPE is again quite higher with the equivalent empirical approach of Turc, which is obtained between 10.1 (at Jagdalpur) to 13.9 (at Raipur).
- iii. Subsequently, for type III models, MAPE values are computed close to that of type II models, which varied between a lowest of 4.7 with RBFDE3 & RBFPSO3 (at Raipur) to a high of 8.0 with MLANN3 (at Jagdalpur). MAPE for the Open Pan approach varies between 12.2 (at Raipur) to 22.2 (at Ambikapur),

Table 5 MAPE (%) and RMSE ($mm\ week^{-1}$) for different types of soft computing and equivalent empirical models with test data sets of Ambikapur, Jagdalpur, and Raipur

Type	MODEL	Ambikapur		Jagdalpur		Raipur	
		MAPE	RMSE	MAPE	RMSE	MAPE	RMSE
I	RBFDE1	8.9*	2.96	10.6	3.63	7.4	2.98
	RBFPSO1	9.3	3.04	10.7	3.67	7.4	2.98
	RBFNN1	9.3	3.06	10.5	3.79	8.5	3.22
	MLANN1	10.0	3.11	11.8	3.88	9.0	3.23
	Hargreaves	30.3	7.15	29.0	7.53	22.6	6.13
II	RBFDE2	4.9	2.10	5.3	2.74	5.7	2.63
	RBFPSO2	5.0	2.10	5.4	2.78	5.9	2.68
	RBFNN2	4.9	2.23	5.2	2.81	5.9	3.14
	MLANN2	5.1	2.31	6.2	2.93	10.2	3.45
	Turc	13.9	3.73	10.1	4.26	13.9	6.54
III	RBFDE3	5.9	1.82	7.0	2.18	4.7	1.75
	RBFPSO3	6.0	1.89	6.9	2.21	4.7	1.77
	RBFNN3	6.3	1.95	7.0	2.27	4.9	1.80
	MLANN3	7.8	2.43	8.0	2.48	6.6	2.28
	Open-Pan	22.2	6.55	21.7	6.30	16.2	5.05
IV	RBFDE4	2.1	0.68	3.7	1.06	1.1	0.36
	RBFPSO4	3.4	1.02	3.8	1.10	1.3	0.43
	RBFNN4	4.4	1.32	4.1	1.22	2.4	0.82
	MLANN4	4.6	1.36	4.4	1.25	3.9	1.29
	Blaney-Criddle	22.1	6.12	15.5	4.27	22.6	7.20
V	RBFDE5	2.2	0.66	3.4	1.06	1.9	0.80
	RBFPSO5	3.9	1.08	3.6	1.08	2.2	0.87
	RBFNN5	4.6	1.38	4.7	1.39	2.9	1.07
	MLANN5	5.2	1.48	5.2	1.46	4.1	1.31

*boldface numbers highlight the best results

Table 6 R² and NSE for different types of soft computing and equivalent empirical models with test datasets of Ambikapur, Jagdalpur, and Raipur

Type	MODEL	Ambikapur		Jagdalpur		Raipur	
		R ²	NSE	R ²	NSE	R ²	NSE
I	RBFDE1	0.949	0.892*	0.878	0.785	0.955	0.932
	RBFPSO1	0.950	0.886	0.880	0.781	0.954	0.933
	RBFNN1	0.950	0.884	0.876	0.766	0.948	0.922
	MLANN1	0.938	0.881	0.871	0.755	0.946	0.921
	Hargreaves	0.933	0.371	0.809	0.077	0.900	0.716
II	RBFDE2	0.981	0.946	0.955	0.950	0.969	0.948
	RBFPSO2	0.980	0.945	0.954	0.948	0.969	0.946
	RBFNN2	0.978	0.939	0.950	0.947	0.946	0.926
	MLANN2	0.976	0.934	0.948	0.939	0.929	0.910
	Turc	0.905	0.829	0.795	0.704	0.833	0.677
III	RBFDE3	0.960	0.959	0.934	0.922	0.978	0.977
	RBFPSO3	0.956	0.956	0.930	0.920	0.977	0.976
	RBFNN3	0.954	0.953	0.931	0.916	0.977	0.976
	MLANN3	0.928	0.927	0.908	0.900	0.961	0.961
	Open-Pan	0.743	0.472	0.827	0.353	0.947	0.808
IV	RBFDE4	0.995	0.994	0.988	0.982	0.999	0.999
	RBFPSO4	0.989	0.987	0.986	0.980	0.999	0.999
	RBFNN4	0.982	0.978	0.985	0.976	0.996	0.995
	MLANN4	0.982	0.977	0.983	0.975	0.990	0.988
	Blaney-Criddle	0.826	0.538	0.827	0.703	0.821	0.608
V	RBFDE5	0.995	0.995	0.983	0.982	0.996	0.995
	RBFPSO5	0.986	0.986	0.983	0.981	0.995	0.994
	RBFNN5	0.980	0.976	0.973	0.969	0.993	0.991
	MLANN5	0.974	0.973	0.974	0.965	0.988	0.987

*boldface numbers highlight the best results

which is very high as compared to type III soft computing models.

- iv. Type IV soft computing models yield better results as compared to all other types of soft computing and empirical models. MAPE ranges between a low of 1.1 to a high of 3.9 at Raipur, followed by 3.7 to 4.4 at Jagdalpur and 2.2 to 4.6 at Ambikapur with RBFDE4 and MLANN4 respectively. MAPE with the Blaney-Criddle method is again quite inferior as compared to type IV soft computing approaches and ranges from 15.5 (at Jagdalpur) to 22.6 (at Raipur).
- v. Type V models also produced good results, as reasonably fair estimates of ET₀ can be obtained between a low MAPE of 1.9 with RBFDE5 (at Raipur) to 5.2 with MLANN5 (at Jagdalpur and Ambikapur), which is very much comparable to that of type IV models, even without taking humidity data as one of the input features.
- vi. Regarding RMSE, type I soft computing models have resulted in RMSE between 2.98 mm week⁻¹ (at Raipur) with RBFDE1 and RBFPSO1 to 3.88 week⁻¹ (at Jagdalpur) with MLANN1, as against the higher RMSE of 6.13 week⁻¹ (at Raipur) to 7.53 mm week⁻¹ (at Jagdalpur) obtained with Hargreaves approaches.
- vii. Type II soft computing models have produced improved RMSE as compared to type I models, which ranges between a low of 2.10 mm week⁻¹ with RBFDE2 (at Ambikapur) to a high of 3.45 mm week⁻¹ with MLANN2 (at Raipur). Interestingly, at Jagdalpur the soft computing models produce comparatively better estimates of FAO56-PM ET₀ in terms of RMSE as compared with similar models at Jagdalpur and Raipur. In general, type II soft computing models have yielded better ET₀ estimates as compared to Turc methods, for which RMSE ranges between 3.73 (at Ambikapur) to 6.54 mm week⁻¹ (at Raipur).
- viii. Regarding type III models, RMSE has improved further and is computed between a low of 1.75 mm week⁻¹ with RBFDE3 to a high of 2.28 mm week⁻¹ with MLANN3 at Raipur, whereas the same for Jagdalpur and Ambikapur, it varied between a low of 1.82 mm week⁻¹ with RBFDE3 to a high of 2.48 mm week⁻¹ with MLANN3. The equivalent empirical method of Open Pan has produced higher RMSE, which varied between 5.05 mm week⁻¹

in Raipur to 6.55 mm week⁻¹ at Ambikapur, which is almost three times more as compared to type III soft computing models.

- ix. Similar to MAPE, type IV soft computing models have yielded excellent results in terms of RMSE also. In Raipur, RMSE ranges between the lowest of 0.36 mm week⁻¹ with RBFDE4 to the highest of 1.29 mm week⁻¹ with MLANN4. At Jagdalpur, it ranges between 1.06 mm week⁻¹ with RBFDE4 to 1.25 mm week⁻¹ with MLANN4, whereas at Ambikapur, RMSE ranges between 0.68 mm week⁻¹ with RBFDE4 to 1.36 mm week⁻¹ with MLANN4. The low RMSE values (< 1 mm week⁻¹) obtained with evo-

lutionary optimized hybrid soft computing models (RBFDE4 and RBFDE5) are quite encouraging. This demonstrates the potential of the RBFDE4 and RBFPSO4 models and these models may consider as an alternative to the FAO56-PM empirical approach for ET₀ estimation in the study area. In contrast, Blaney-Cridde has produced very high RMSE, which ranges between 4.27 to 7.29 mm/week at different locations, similar to that of the Open Pan method.

- x. Type V, soft computing models have also produced better results which is quite identical with type IV models even without including humidity data as an input feature. RMSE with type V models ranges

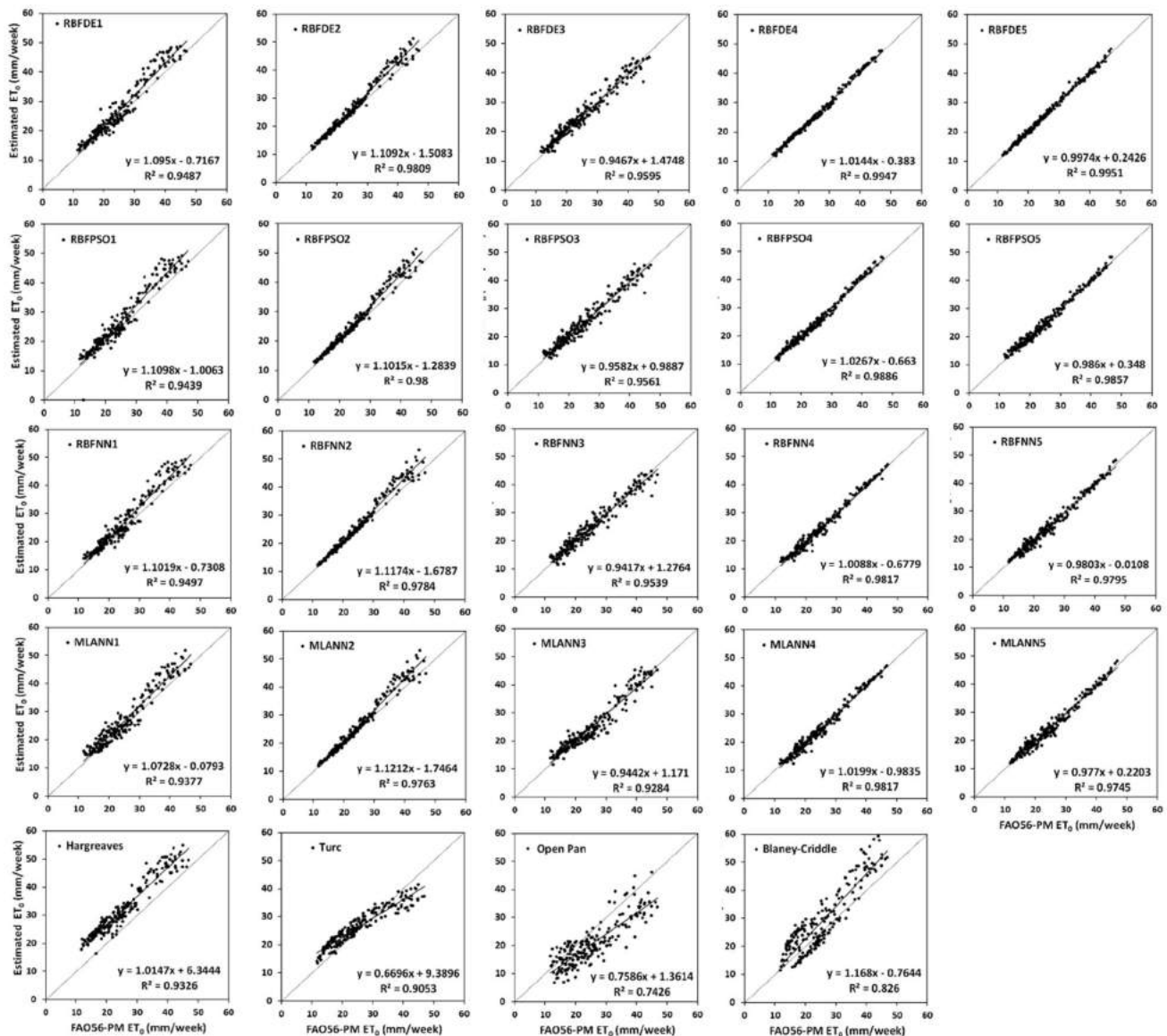


Fig. 6 Relationship between estimated ET₀ and FAO56-PM ET₀ for different soft computing and empirical models with test data sets at Ambikapur

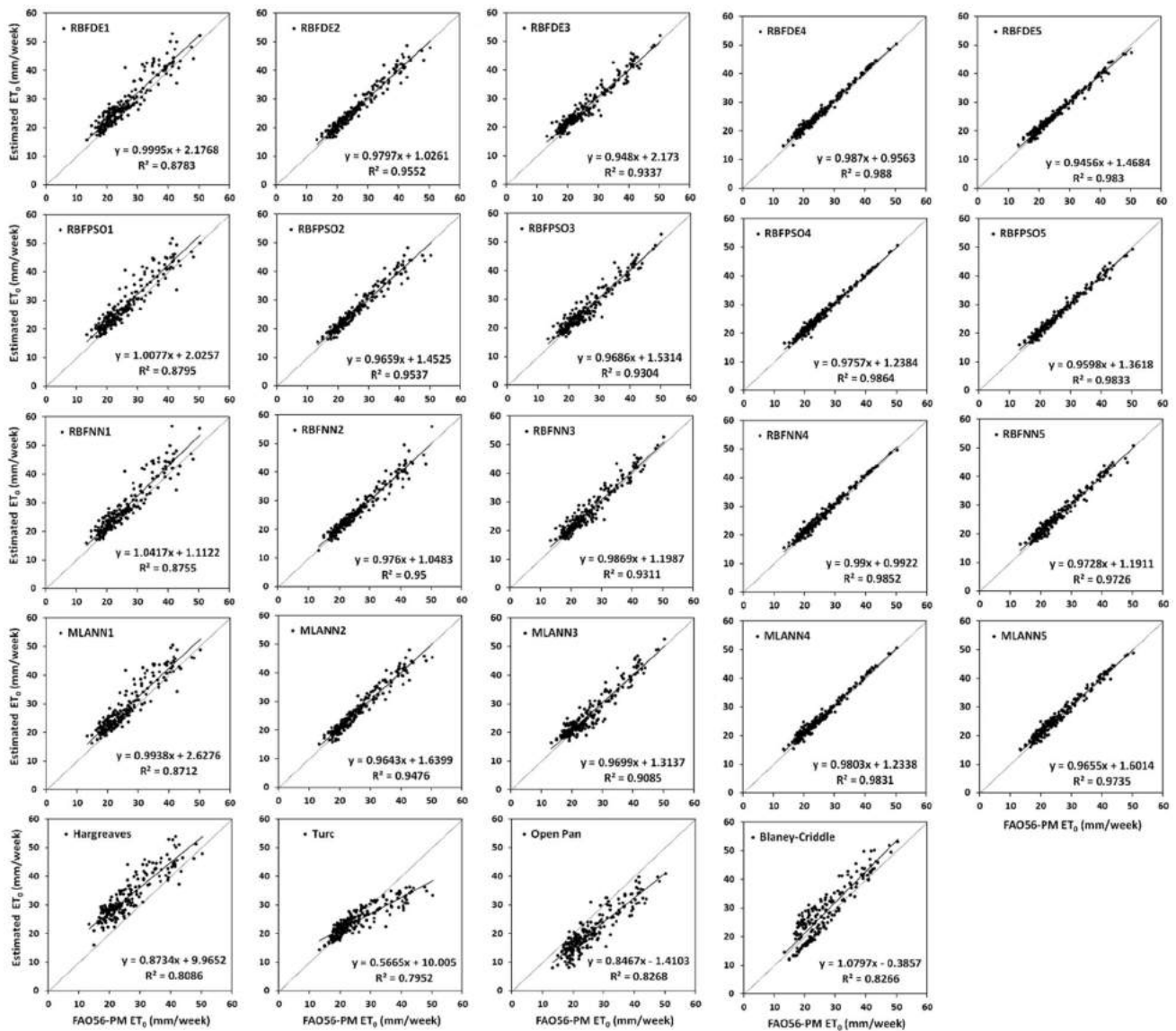


Fig. 7 Relationship between estimated ET_0 and FAO56-PM ET_0 for different soft computing and empirical models with test data sets at Jagdalpur

between lowest $0.66 \text{ mm week}^{-1}$ with RBFDE5 to highest of $1.48 \text{ mm week}^{-1}$ with MLANN5 at Ambikapur. At Raipur and Jagdalpur, RMSE ranges between 0.80 to 1.31 and 1.06 to $1.46 \text{ mm week}^{-1}$ with RBFDE5 and MLANN5 models respectively.

To further examine the relationship between the estimated and FAO56-PM ET_0 , two more statistical measures, R^2 and NSE are computed for different soft computing and empirical models and shown in Table 6. The linear relationship between estimated ET_0 and FAO56- ET_0 is also depicted in Figs. 6, 7, and 8 in Ambikapur, Jagdalpur, and Raipur respectively. In general, both R^2 and NSE convey similar information about the model performance and therefore,

the marginal difference is observed between these two performance evaluation measures within a similar type of model in different locations. However, sometimes R^2 values give a false indication and produce higher values close to 1 despite a very high intercept. In such cases corresponding NSE helps in evaluating the model performance.

- xi. Type IV soft computing models have produced better R^2 and NSE values as compared to all other models considered for investigation. The highest R^2 values of 0.999, 0.988, and 0.995 are obtained with RBFDE4 and RBFPSO4 in Raipur, Jagdalpur, and Ambikapur respectively with test data sets. The remaining type IV soft computing models also pro-

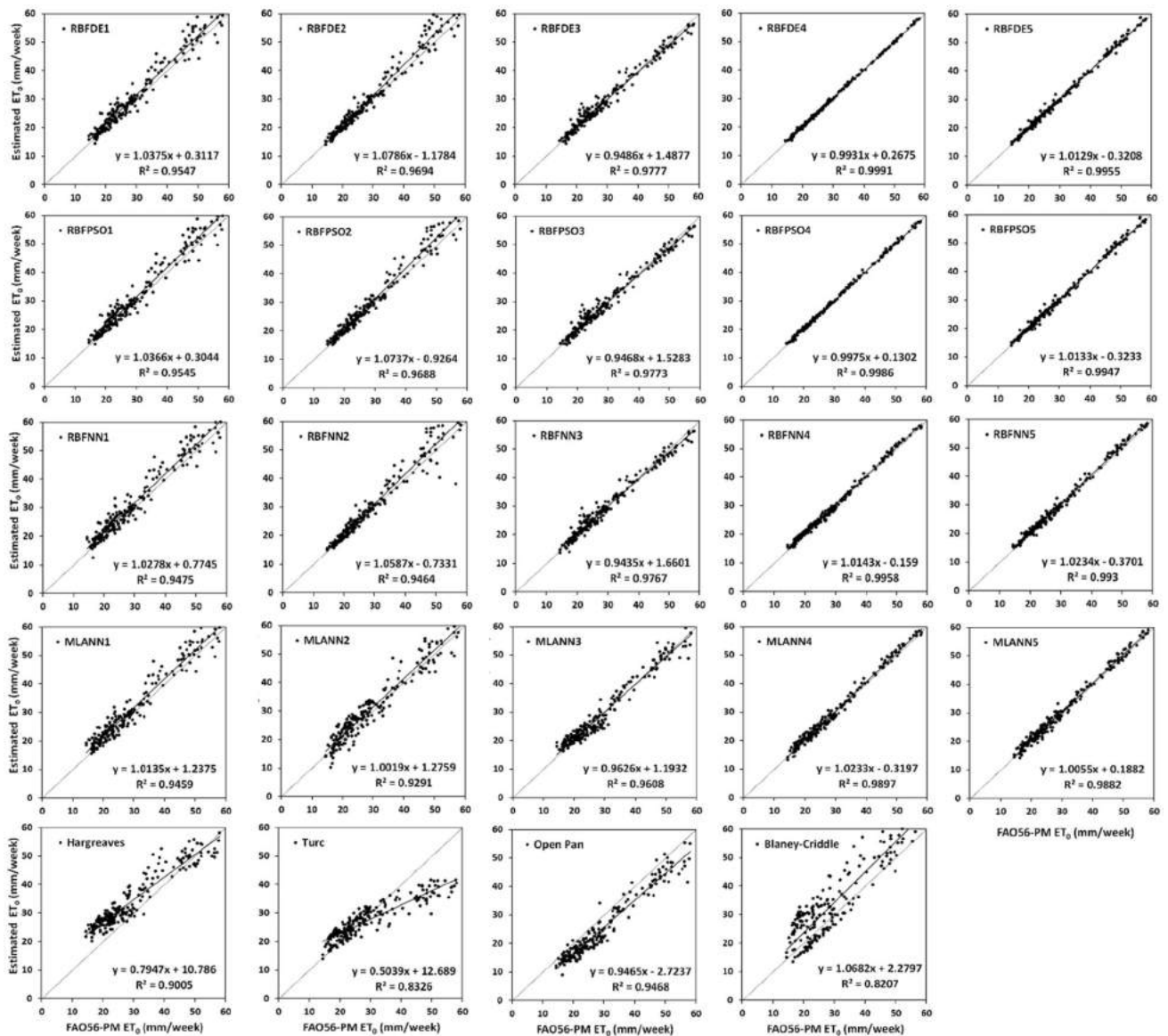


Fig. 8 Relationship between estimated ET₀ and FAO56-PM ET₀ for different soft computing and empirical models with test data sets at Raipur

duced good R² and NSE values as compared to the equivalent empirical approach.

- xii. Similar results are obtained with type V soft computing models, as R² and the corresponding NSE vary between 0.965 to 0.996 in different locations, with RBFDE5 and RBFPSO5 being the best models.
- xiii. For the type III model, R² and NSE range between 0.961 to 0.978 at Raipur and between 0.900 to 0.960 at Jagdalpur and Ambikapur. R² and NSE with type II soft computing models vary between 0.910 to 0.981. Hence, it can be stated that consistent ET₀ estimates with a fair degree of agreement between estimated

and target ET₀ can be obtained using Type both II and III soft computing models.

- xiv. Type I soft computing models of RBFDE1 and RBFPSO1 have resulted in slightly lower R² and NSE values than RBFNN1 and MLANN1 as compared to remaining types mainly as fewer input features are involved in computations.
- xv. Inconsistence and low R² and NSE values that have obtained with empirical approaches of Hargreaves, Turc, Open Pan, and Blaney Criddle as compared to their equivalent soft computing models of respective types, clearly establish the fact that soft comput-

ing models produce better estimates of FAO56-PM ET_0 than empirical models.

Results of the performance evaluation analysis indicate that the evolutionary optimized hybrid soft computing models considered for the investigation (RBFDE and RBFPSO) performed consistently better than other conventional soft computing techniques (RBFNN and MLANN) and empirical approaches in all the objectives. From the inferences, it is also evident that when a complete set of the climatic variable is involved in the computation of ET_0 using these models, it looks very difficult to choose between RBFDE and RBFPSO as they look statistically similar in some cases. However, the proposed RBFDE is recommended because of its preciseness and generalization performance in estimating ET_0 in all the stations considered for the study.

4 Conclusions ET_0

The present investigation is carried out to examine the generalized potential of evolutionary optimized hybrid soft computing techniques RBFDE and RBFPSO for the estimation of ET_0 in different ACZs. The ET_0 estimates obtained with proposed RBFDE and RBFPSO models are compared to the conventional neural network (RBFNN, MLANN) and existing empirical approaches. Looking to the scarcity of complete datasets required for computation of FAO-PM ET_0 , four variants of each category of soft computing models (RBFDE, RBFPSO, RBFNN, and MLANN) equivalent (in terms of input feature combination) to empirical approaches (Hargreaves, Turc, Open Pan and Blaney-Criddle) is examined. It can be concluded that different soft computing models considered in this investigation, have resulted in improved and more consistent FAO56-PM ET_0 estimates as compared to equivalent empirical approaches. Among the soft computing models, evolutionary models RBFDE and RBFPSO produced a more precise estimation of FAO56-PM ET_0 than conventional RBFNN and MLANN as proposed RBFDE and RBFPSO models resulted in low MAPE and RMSE and high R^2 and NSE close to 1 in most of the cases. However, ET_0 estimates obtained with the proposed RBFDE seems to be slightly better than RBFPSO. Hence, appropriate soft computing models may be recommended for the estimation of ET_0 in other stations of respective ACZs of the study area. The proposed soft computing models may be embedded in crop weather simulation models as subroutines for precise estimation ET_0 with available input features. However, re-calibration and re-validation of these data-driven models are essentially required for their effective implantation in other parts of the world.

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest.

Ethical approval This article does not contain any studies with human participants or animals performed by any of the authors.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

1. Allen RG, Pereira LS, Raes D, Smith M (1998) Introduction to evapotranspiration. In: Crop evapotranspiration-guidelines for computing crop water requirements-FAO Irrigation and drainage paper, vol 56, pp 1–13. FAO, Rome. <https://doi.org/10.1016/j.eja.2010.12.001>
2. Elizabeth AH, Robert EC (2013) Water balance estimates of evapotranspiration rates in areas with varying land use. In: Evapotranspiration-an overview. InTech. <https://doi.org/10.5772/52811>
3. Allen RG, Jensen ME, Wright JL, Burman RD (1989) Operational estimates of reference evapotranspiration. *Agron J* 81(4):650–662. <https://doi.org/10.2134/agronj1989.00021962008100040019x>
4. Droogers P, Allen RG (2002) Estimating reference evapotranspiration under inaccurate data conditions. *Irrigat Drain Syst* 16(1):33–45. <https://doi.org/10.1023/A:1015508322413>
5. Hargreaves GL, Hargreaves GH, Riley JP (1985) Agricultural, benefits for Senegal River Basin. *J Irrig Drain Eng ASCE* 111:113–124
6. Turc L (1961) Evaluation des besoins en eau d'irrigation, evapotranspiration potentielle, formule climatique simplifiée et mise a jour. *Ann Agron* 12:13–49 (in French)
7. Doorenbos J, Pruitt WO (1977) Guidelines for predicting crop water requirements. Irrigation and drainage paper No 24, 2nd edn, Food and Agriculture Organization, Rome, p 156
8. Christiansen JE (1968) Pan evaporation and evapotranspiration by climatic data. *J Irrig Drain Div Am Soc Civil Eng* 94:243–263
9. Bapuji RB, Sandeep VM, Rao VUM, Venkateswarlu B (2012) Potential Evapotranspiration estimation for Indian conditions: Improving accuracy through calibration coefficients. *Technical Bull No 1/2012*. All India Co-ordinated Research Project on Agrometeorology, Central Research Institute for Dryland Agriculture, Hyderabad, p 60
10. Chauhan S, Shrivastava RK (2009) Performance evaluation of reference evapotranspiration estimation using climate based methods and artificial neural networks. *Water Resour Manag* 23(5):825–837

11. Kisi O, Guven A (2010) Evapotranspiration modeling using linear genetic programming technique. *J Irrig Drain Eng* 136(10):715–723. [https://doi.org/10.1061/\(asce\)ir.1943-4774.0000244](https://doi.org/10.1061/(asce)ir.1943-4774.0000244)
12. Kumar M, Raghuwanshi NS, Singh R (2011) Artificial neural networks approach in evapotranspiration modeling: a review. *Irrig Sci*. <https://doi.org/10.1007/s00271-010-0230-8>
13. Mallikarjuna P, Jyothy SA, Sekhar Reddy KC (2012) Daily reference evapotranspiration estimation using linear regression and ANN models. *J Inst Eng Ser A* 93(4):215–221. <https://doi.org/10.1007/s40030-013-0030-2>
14. Kisi O (2016) Modeling reference evapotranspiration using three different heuristic regression approaches. *Agric Water Manag* 169:162–172. <https://doi.org/10.1016/j.agwat.2016.02.026>
15. Yassin MA, Alazba AA, Mattar MA (2016) Artificial neural networks versus gene expression programming for estimating reference evapotranspiration in an arid climate. *Agric Water Manag* 163:110–124. <https://doi.org/10.1016/j.agwat.2015.09.009>
16. Antonopoulos VZ, Antonopoulos AV (2017) Daily reference evapotranspiration estimates by artificial neural networks technique and empirical equations using limited input climate variables. *Comput Electron Agric* 132:86–96. <https://doi.org/10.1016/j.compag.2016.11.011>
17. Nema MK, Khare D, Chandniha SK (2017) Application of artificial intelligence to estimate the reference evapotranspiration in sub-humid Doon valley. *Appl Water Sci* 7(7):3903–3910. <https://doi.org/10.1007/s13201-017-0543-3>
18. Pandey PK, Nyori T, Pandey V (2017) Estimation of reference evapotranspiration using data-driven techniques under limited data conditions. *Model Earth Syst Environ* 3(4):1449–1461. <https://doi.org/10.1007/s40808-017-0367-z>
19. Patil AP, Deka PC (2017) Performance evaluation of hybrid Wavelet-ANN and Wavelet-ANFIS models for estimating evapotranspiration in arid regions of India. *Neural Comput Appl* 28(2):275–285. <https://doi.org/10.1007/s00521-015-2055-0>
20. Wen X, Si J, He Z, Wu J, Shao H, Yu H (2015) Support-vector-machine-based models for modeling daily reference evapotranspiration with limited climatic data in extreme arid regions. *Water Resour Manag* 29(9):3195–3209. <https://doi.org/10.1007/s11269-015-0990-2>
21. Partal T (2016) Comparison of wavelet-based hybrid models for daily evapotranspiration estimation using meteorological data. *KSCE J Civil Eng* 20(5):2050–2058. <https://doi.org/10.1007/s12205-015-0556-0>
22. Kisi O, Demir V (2016) Evapotranspiration estimation using six different multi-layer perceptron algorithms. *Irrig Drain Syst Eng*. <https://doi.org/10.4172/2168-9768.1000164>
23. Dou X, Yang Y (2018) Evapotranspiration estimation using four different machine learning approaches in different terrestrial ecosystems. *Comput Electron Agric* 148:95–106. <https://doi.org/10.1016/j.compag.2018.03.010>
24. Adamala S (2018) Temperature based generalized wavelet-neural network models to estimate evapotranspiration in India. *Inf Process Agric* 5(1):149–155. <https://doi.org/10.1016/j.inpa.2017.09.004>
25. Sanikhani H, Kisi O, Maroufpoor E, Yaseen ZM (2018) Temperature-based modeling of reference evapotranspiration using several artificial intelligence models: application of different modeling scenarios. *Theoret Appl Climatol* 135(1–2):449–462. <https://doi.org/10.1007/s00704-018-2390-z>
26. Ozkan C, Kisi O, Akay B (2011) Neural networks with artificial bee colony algorithm for modeling daily reference evapotranspiration. *Irrig Sci* 29(6):431–441. <https://doi.org/10.1007/s00271-010-0254-0>
27. Eslamian SS, Gohari SA, Zareian MJ, Firoozfar A (2012) Estimating Penman-Monteith reference evapotranspiration using artificial neural networks and genetic algorithm: a case study. *Arab J Sci Eng* 37(4):935–944. <https://doi.org/10.1007/s13369-012-0214-5>
28. Aghajanloo MB, Sabziparvar AA, Hosseinzadeh TP (2013) Artificial neural network-genetic algorithm for estimation of crop evapotranspiration in a semi-arid region of Iran. *Neural Comput Appl* 23(5):1387–1393. <https://doi.org/10.1007/s00521-012-1087-y>
29. Petković D, Gocić M, Shamshirband S, Qasem SN, Trajković S (2016) Particle swarm optimization-based radial basis function network for estimation of reference evapotranspiration. *Theoret Appl Climatol* 125(3–4):555–563. <https://doi.org/10.1007/s00704-015-1522-y>
30. Jovic S, Nedeljkovic B, Golubovic Z, Kostic N (2018) Evolutionary algorithm for reference evapotranspiration analysis. *Comput Electron Agric* 150:1–4. <https://doi.org/10.1016/j.compag.2018.04.003>
31. Feng Y, Cui N, Zhao L, Hu X, Gong D (2016) Comparison of ELM, GANN, WNN and empirical models for estimating reference evapotranspiration in the humid region of Southwest China. *J Hydrol* 536:376–383. <https://doi.org/10.1016/j.jhydrol.2016.02.053>
32. Gocić M, Motamedi S, Shamshirband S, Petković D, Ch S, Hashim R, Arif M (2015) Soft computing approaches for forecasting reference evapotranspiration. *Comput Electron Agric* 113:164–173. <https://doi.org/10.1016/j.compag.2015.02.010>
33. Mehdizadeh S, Behmanesh J, Khalili K (2017) Using MARS, SVM, GEP and empirical equations for estimation of monthly mean reference evapotranspiration. *Comput Electron Agric* 139:103–114. <https://doi.org/10.1016/j.compag.2017.05.002>
34. Mattar MA, Alazba AA (2018) GEP and MLR approach for the prediction of reference evapotranspiration. *Neural Comput Appl*. <https://doi.org/10.1007/s00521-018-3410-8>
35. Baltzis K (2013) Patented applications of differential evolution in microwave and communication engineering. *Recent Patents Comput Sci* 6(2):115–128. <https://doi.org/10.2174/22132759113069990004>
36. Tenaglia GC, Lebensztajn L (2014) A multiobjective approach of differential evolution optimization applied to electromagnetic problems. *IEEE Trans Magn*. <https://doi.org/10.1109/TMAG.2013.2285980>
37. Uher V, Gajdoš P, Radecký M, Snášel V (2016) Utilization of the discrete differential evolution for optimization in multidimensional point clouds. *Comput Intell Neurosci*. <https://doi.org/10.1155/2016/6329530>
38. Das S, Abraham A, Konar A (2008) Particle swarm optimization and differential evolution algorithms: technical analysis, applications and hybridization perspectives. *Stud Comput Intell* 116:1–38. https://doi.org/10.1007/978-3-540-78297-1_1
39. Hui S, Suganthan PN (2016) Ensemble and arithmetic combination-based speciation differential evolution for multimodal optimization. *IEEE Trans Cybern* 46(1):64–74. <https://doi.org/10.1109/TCYB.2015.2394466>
40. Ramadas M, Abraham A, Kumar S (2018) RDE reconstructed mutation strategy for differential evolution algorithm. *Adv Intell Syst Comput* 614:76–85. https://doi.org/10.1007/978-3-319-60618-7_8
41. Ramadas M, Pant M, Abraham A, Kumar S (2019) Segmentation of weather radar image based on hazard severity using RDE: reconstructed mutation strategy for differential evolution algorithm. *Neural Comput Appl* 31:1253–1261. <https://doi.org/10.1007/s00521-017-3091-8>
42. Ghosh A, Mallipeddi R, Das S, Das AK (2018) A switched parameter differential evolution with multi-donor mutation and annealing based local search for optimization of lennard-jones atomic clusters. In: 2018 IEEE congress on evolutionary computation,

- CEC 2018-Proceedings. Institute of Electrical and Electronics Engineers Inc. <https://doi.org/10.1109/CEC.2018.8477991>
43. Biswas PP, Suganthan PN, Mallipeddi R, Amaratunga GAJ (2018) Optimal power flow solutions using differential evolution algorithm integrated with effective constraint handling techniques. *Eng Appl Artif Intell* 68:81–100. <https://doi.org/10.1016/j.engap.2017.10.019>
 44. Rout M, Majhi B, Majhi R, Panda G (2014) Forecasting of currency exchange rates using an adaptive ARMA model with differential evolution based training. *J King Saud Univ Comput Inf Sci* 26(1):7–18. <https://doi.org/10.1016/j.jksuci.2013.01.002>
 45. Pujari P, Majhi B (2017) Application of nature-inspired technique to Odia handwritten numeral recognition, book chapter in handbook of research on modeling, analysis and application of nature-inspired Metaheuristic algorithms, IGI Global Publication, USA, pp 377–399 <https://doi.org/10.4018/978-1-5225-2857-9.ch019>
 46. World Meteorological Organization (2012) WMO-No. 8-Guide to meteorological instruments and methods of observation, pp I.8-1–I.9-1
 47. Broomhead D, Lowe D (1988) Multivariable functional interpolation and adaptive networks. *Complex Syst* 2:321–355
 48. Fernández-Redondo M, Hernández-Espinosa C, Ortiz-Gómez M, Torres-Sospedra J (2004) Training radial basis functions by gradient descent. In: *Lecture notes in artificial intelligence (subseries of lecture notes in computer science)*, Vol 3070, pp 184–189. Springer Verlag. https://doi.org/10.1007/978-3-540-24844-6_23
 49. Storn R, Price K (1995) Differential evolution—a simple and efficient adaptive scheme for global optimization over continuous spaces. Technical Report TR-95-012, pp 1–12 <https://doi.org/10.1.1.1.9696>
 50. Storn R, Price K (1997) Differential evolution—a simple and efficient Heuristic for global optimization over continuous spaces. *J Global Optim* 11(4):341–359. <https://doi.org/10.1023/A:1008202821328>
 51. Kennedy J, Eberhart R (1995) Particle swarm optimization. *Proc IEEE Int Conf Neural Netw* 4:1942–1948
 52. Eberhart R, Kennedy J (2002) A new optimizer using particle swarm theory. In: *Institute of Electrical and Electronics Engineers (IEEE)*, pp 39–43 <https://doi.org/10.1109/mhs.1995.494215>
 53. Eberhart R, Yuhui S (2002) Particle swarm optimization: developments, applications and resources. In: *Institute of Electrical and Electronics Engineers (IEEE)*, pp 81–86 <https://doi.org/10.1109/cec.2001.934374>
 54. del Valle Y, Venayagamoorthy GK, Mohagheghi S, Hernandez JC, Harley RG (2008) Particle swarm optimization: basic concepts, variants and applications in power systems. *IEEE Trans Evol Comput*. <https://doi.org/10.1109/TEVC.2007.896686>
 55. Chen LF, Su CT, Chen KH (2012) An improved particle swarm optimization for feature selection. *Intell Data Anal* 16(2):167–182. <https://doi.org/10.3233/IDA-2012-0517>
 56. Zhang Y, Wang S, Ji GA (2015) Comprehensive survey on particle Swarm optimization algorithm and its applications. *Mathematical Problems in Engineering*, pp 1–38 <https://doi.org/10.1155/2015/931256>
 57. Haykin S (1998) *Neural networks—a comprehensive foundation*, 2nd edn. Prentice-Hall, Upper Saddle River, pp 26–32
 58. Nash JE, Sutcliffe JV (1970) River flow forecasting through conceptual models part I—a discussion of principles. *J Hydrol* 10(3):282–290
- Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Modelling economic policy issues

Impact of COVID-19 on GDP of major economies: Application of the artificial neural network forecaster

Pradyot Ranjan Jena ^{a,*}, Ritanjali Majhi ^a, Rajesh Kalli ^b, Shunsuke Managi ^c, Babita Majhi ^d

^a School of Management, National Institute of Technology Karnataka, Surathkal, Mangalore 575025, India

^b Department of Management Studies, Madanapalle Institute of Technology & Science, India

^c Urban Institute & Department of Civil Engineering, Kyushu University, Japan

^d Department of CSIT, Guru Ghasidas Vishwavidyalaya, Central University, Bilaspur, India



ARTICLE INFO

Article history:

Received 8 August 2020

Received in revised form 9 December 2020

Accepted 10 December 2020

Available online 18 December 2020

Keywords:

Forecasting

GDP

Multilayer neural network

Mean absolute percentage error

ABSTRACT

The ongoing COVID-19 pandemic has caused global health impacts, and governments have restricted movements to a certain extent. Such restrictions have led to disruptions in economic activities. In this paper, the GDP figures for the April–June quarter of 2020 for eight countries, namely, the United States, Mexico, Germany, Italy, Spain, France, India, and Japan, are forecasted. Considering that artificial neural network models have higher forecasting accuracy than statistical methods, a multilayer artificial neural network model is developed in this paper. This model splits the dataset into two parts: the first with 80% of the observations and the second with 20%. The model then uses the first part to optimize the forecasting accuracy and then applies the optimized parameters to the second part of the dataset to assess the model performance. A forecasting error of less than 2% is achieved by the model during the testing procedure. The forecasted GDP figures show that the April–June quarter of the current year experienced sharp declines in GDP for all countries. Moreover, the annualized GDP growth is expected to reach double-digit negative growth rates. Such alarming prospects require urgent rescue actions by governments.

© 2020 Economic Society of Australia, Queensland. Published by Elsevier B.V. All rights reserved.

1. Introduction

The novel coronavirus disease 2019 (COVID-19), which first appeared in Wuhan city, China, in December 2019, has caused global distress, claiming lives and collapsing economies, as many individuals are connected globally (Acemoglu et al., 2020; Nakamura and Managi, 2020). Given its deepening threat to human lives and economies, the Director-General of the World Health Organization (WHO) declared COVID-19 a Public Health Emergency of International Concern (WHO, 2020) on 30th January 2020. Policymakers in every country are under pressure to maintain a balance between containing the disease by implementing lockdowns and saving the jobs and livelihoods of a large number of people by keeping economic activities undeterred (Yoo and Managi, 2020). Restrictions on the movements of both people and goods have disrupted supply chains and accelerated the unemployment problem. Given this consideration, it has become important for countries to assess the broad economic implications of COVID-19. Macroeconomic indicators represent the health and stability of a country's economy. Gross domestic product (GDP), the most widely recognized indicator, accounts for the

* Corresponding author.

E-mail address: pradyotjena@nitk.edu.in (P.R. Jena).

overall goods and services produced within a country. Accurate GDP projections can equip policymakers with tools to effectively plan for future economic development. In this paper, we develop a multilayer neural network model that can forecast GDP with minimum error.

Previous methodologies employed in GDP forecasting can be classified into two broad categories. The first emphasizes a strong theoretical background and applies linear models (Guégan and Rakotomaroahy, 2010). However, concerns with this approach have been raised due to strong hypotheses on model specification, estimation, and asymptotic properties of the estimated parameters (Guégan and Rakotomaroahy, 2010). The second, nonlinear models, includes the nearest-neighbors method and neural network (Härdle et al., 2012; Tkacz, 2001a; Kock and Teräsvirta, 2014). The nearest-neighbors method uses parametric models and neural networks for nonparametric modeling. Forecasting GDP using a linear autoregressive model based on the Box–Jenkins approach or multivariate VAR (Box et al., 2015; Bařbura et al., 2010; Balcilar et al., 2015) is common. Additionally, artificial neural networks (ANNs) have been applied to forecast macroeconomic indicators such as inflation, exchange rates, oil prices, and interest rates (Hlaváček et al., 2005; Jena et al., 2015; Ali Choudhary and Haider, 2012; McNelis, 2005; Ramos-Pérez et al., 2019; Szafranek, 2019). Furthermore, ANN models have been used to forecast GDP in different countries, such as the United States (Loermann and Maas, 2019), China (Shi et al., 2006), Sweden (Teräsvirta, 2005), Romania (Saman, 2011), and Canada (Tkacz, 2001b). Recently, Torres and Qiu (2018) employed the ANN method to forecast returns from several cryptocurrencies, exchange rates, commodities, and stocks.

Several studies have demonstrated that ANN models yield more accurate predictions than econometric models (Tkacz, 2001a,b; Ali Choudhary and Haider, 2012; McNelis and McAdam, 2004). Shi et al. (2006) used a neural network with a genetic algorithm to forecast China's GDP. The quarterly data employed in the model yielded accurate and efficient estimates. Jahn (2020) demonstrated that an ANN model yielded better performance than a linear model in predicting the annual GDP of 15 industrialized economies. Furthermore, Chuku et al. (2019) estimated the GDP for South Africa, Kenya, and Nigeria and found that an ANN model was superior to traditional econometric models and ARIMA. With increased exposure to chaotic influences (political factors, external factors, and commodity prices) among different countries, linear models are becoming less suitable; ANN models are more accurate given their flexibility in modeling (Chuku et al., 2019).

Uncertainty is common in macroeconomic activity, which is a challenge for researchers keen on predicting the future of such indicators in a constantly changing environment. During the current pandemic period, concerns have been raised about a future global economic crisis. The International Monetary Fund (IMF) projects a 4.9% decline in global economic growth for 2020–21: an 8% decline for advanced economies, and 3% and 1% declines for emerging and low-income economies, respectively (International Monetary Fund, 2020). Similar estimates from the OECD indicated a 0.5 to 1.5% decline in global economic growth, while the World Bank and Asian Development Bank predicted 2.1% to 3.9% and 2.3% to 4.8% declines in global economic growth, respectively (RBI, 2020a,b). The increased spread of the pandemic has led many researchers to estimate the economic consequences of the pandemic outbreak. The studies employing annual data are not well suited for this purpose, as they focus on long-term forecasting, whereas policy measures must be devised in the short term. Although a pandemic outbreak usually has both long- and short-term consequences, short-term analysis is more effective from a policymaking perspective. Quantifying the short-term consequences will lead to better decision making to overcome the tragedies associated with the pandemic.

Against this backdrop, we developed a multilayer ANN model to accurately predict quarterly GDP figures for 8 major economies: the United States, Mexico, Germany, Italy, Spain, France, India, and Japan. These countries represent three continents and have experienced a massive onslaught of COVID-19 spread. The main contribution of this paper to the forecasting literature is that it provides a well-calibrated nonlinear model that can accurately predict the impact of a pandemic such as COVID-19. The adaptive model is suitable for analyzing the economic impact of COVID-19. Such predictive models will provide policymakers with a framework to readjust and reinvigorate their economies and can be used for predictions in other countries, as well as the global GDP.

The advantages of the ANN model (Sivanandam and Deepa, 2013) are as follows:

(1) Adaptation potential: The model can learn a pattern or predict a value by adjusting the weights of connections between neurons of different layers.

(2) Self organization: The model organizes itself during the training phase to subsequently perform the desired task.

(3) Real-time operation: After a satisfactory training phase, the model can provide fast prediction and classification performance because of the parallel operations of the artificial neurons of all layers.

(4) Fault tolerance: Because of the massively interconnected parallel network, the ANN performs satisfactorily even after a small amount of damage to the network.

The rest of this paper is organized as follows: Section 2 develops the ANN forecasting model; the simulation procedure of the model is explained in Section 3; the simulation results are presented in Section 4; a discussion of the results is provided in Section 5; finally, the last section concludes with broad policy messages.

2. Development of the Multilayer Artificial Neural Network (MLANN)-based GDP prediction model

Statistical models are inappropriate for prediction when the data are highly nonlinear, uncorrelated, nonstationary, and chaotic, (Teräsvirta, 2006). Nonlinear models such as ANNs are required to circumvent this situation. The MLANN is a multilayered, fully connected, continuously differentiable, nonlinear network that is an appropriate choice to handle the nonlinearity in data (Haykin, 2009). The steps used in the development of the MLANN-based GDP prediction model are shown in Fig. 1.

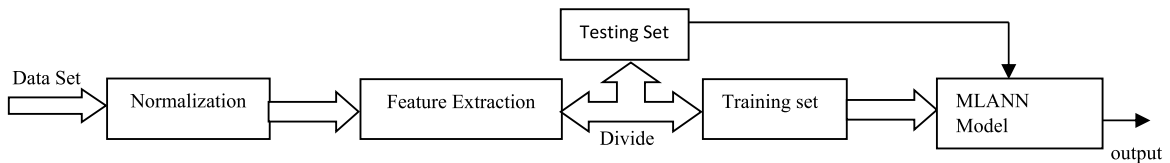


Fig. 1. Steps followed in the development of the GDP prediction model.

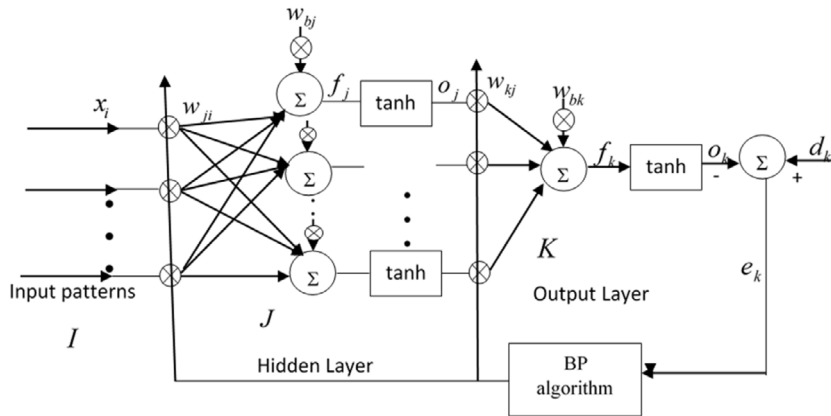


Fig. 2. An MLANN-based GDP prediction model.

The detailed architecture of the MLANN-based prediction model is shown in Fig. 2. Let I, J and K be indices for the input, hidden and output layers, respectively. I represents the number of inputs in each input pattern, J denotes the number of neurons in the hidden layer, and K represents the number of neurons in the output layer. In this case, since the output is one, $K = 1$. Let there be N input patterns: the i th input pattern is represented as x_i . Every input pattern is applied to the input layer of the MLANN model sequentially, weighted, summed, and passed through the activation function (\tanh) to give the output at the hidden layer. The same process continues for the next layer, and the final output of the feedforward network is obtained as o_k . The output is then compared with the desired value or target value, d_k , to calculate the error. This error value is used along with backpropagation (Haykin, 2009) learning to update the weight and bias values of the network. The process continues until the squared error is minimized. The ANN is an adaptive model that iteratively learns from past data during its development stage. Iterative learning means that the connecting weight of each branch updates its old value during the training phase so that the overall training error progressively decreases as the number of iterations increases. The weights are adjusted from their initial random values following a weight update rule known as the back propagation (BP) algorithm. The weight update rules of the BP algorithm are given in Eqs. (6) to (11). The initial weights are chosen randomly; then, the weights of each path of the network are updated using the old weight values, inputs to the weights, back propagated error and learning coefficient. The update process continues until the optimal weights are obtained. The detailed equations for the calculation of the output at each layer and weight update rules are given below.

Referring to the above figure, the output of the k th output neuron o_k is given as

$$o_k = \tanh(f_k) \tag{1}$$

where

$$f_k = \sum_{j=1}^J o_j w_{kj} + w_{bk} \tag{2}$$

o_j = output at the j th hidden neuron

w_{kj} = weights between the j th hidden neuron and k th output neuron

w_{bk} = bias at the k th output neuron

Similarly, the output at the j th hidden layer, o_j , is calculated as

$$o_j = \tanh(f_j) \tag{3}$$

where

$$f_j = \sum_{i=1}^I x_i w_{ji} + w_{bj} \tag{4}$$

x_i = *ith input pattern*
 w_{ji} = *weights between ith input and jth hidden neuron*
 w_{bj} = *bias at jth hidden neuron*

The output of the prediction model o_k is compared with the corresponding target value d_k to determine the error. Hence,

$$e_k = d_k - o_k \tag{5}$$

The weights between the hidden layer and output layer, w_{kj} , are updated using

$$w_{kj} = w_{kj} + \mu * \delta_k * o_j \tag{6}$$

where

$$\delta_k = e_k * \frac{(1 - o_k^2)}{2} \tag{7}$$

μ = *learning parameter lies between 0 to 1*

The bias weight, whose input is always one, is updated as

$$w_{bk} = w_{bk} + \mu * \delta_k \tag{8}$$

Similarly, the weights between the input layer and the hidden layer, w_{ji} , are updated using

$$w_{ji} = w_{ji} + \mu * \delta_j * x_i \tag{9}$$

where

$$\delta_j = \delta_k * w_{kj} * \frac{(1 - o_j^2)}{2} \tag{10}$$

The bias weight of the *jth* neuron in the hidden layer is updated as

$$w_{bj} = w_{bj} + \mu * \delta_j \tag{11}$$

Eqs. (1)–(11) are the key equations for developing the MLANN-based GDP prediction model.

3. Simulation procedure

The simulation of the MLANN-based GDP prediction model is conducted referring to Fig. 2 and using MATLAB 2016 software. The steps are explained in the following section.

(i) Data collection and normalization: Quarterly GDP data were collected for Japan, China, Germany, Spain, France, Italy, the USA, and Mexico from <https://fred.stlouisfed.org>, Economic Research Division, Federal Reserve Bank of St. Louis (FRED | St. Louis Fed, 2020). For India, the data were collected from the Reserve Bank of India (RBI, 2020a,b). The time period of the data and details about the data are provided in Table 1. The GDP data are normalized to a range of 0–1 by dividing each observation by the maximum value of the distribution. Large differences between observations in the data cause problems during modeling calibration, and normalized data help overcome this problem during the calibration of the model and increase the convergence speed.

(ii) Feature extraction: Normalization of the data is followed by feature extraction using a sliding window of size three to generate data patterns or feature patterns. The window is moved over the entire data with a shift of one, and each time, a group of three values is obtained. If N is the length of the data, then there are $N - 2$ groups. For the n th group, $x(n)$, $x(n + 1)$, $x(n + 2)$ are the available values. Every data or input pattern consists of five values, i.e., three values of the group and the slopes between the 1st and 3rd values and 2nd and 3rd values. Mathematically, these values can be represented as: $\{x(n), x(n + 1), x(n + 2), (x(n + 2) - x(n + 1))/x(n + 2), (x(n + 2) - x(n))/x(n + 2)\}$. Hence, at each time point, five inputs are fed to the MLANN model in one input pattern, and there are $N - 2$ patterns in total. Since MLANN is a supervised learning-based model, the target value is also known and stored. For the n th input data pattern, $x(n + 3)$ is the required desired value or target value. The total number of patterns generated for each of the datasets is given in Table 1. Of the total data patterns or input patterns generated, 80% are used to train the model, and the remaining 20% are used to test the model.

(iii) Training of the model: The development of an MLANN-based GDP prediction model is formulated as an optimization problem, where the error between the target value and model estimated values is minimized towards zero. Once the error is nearly zero, the model can predict the correct value of the GDP. A 9:3:1 MLANN structure is used in this paper to

Table 1
Details of the data.

Sl. no.	Name of country	Length of time-series	Length of data	Total number of patterns extracted	Total no. of training patterns	Total number of testing patterns
1	USA	01/01/1990 to 01/10/2020	121	119	97	22
2	Mexico	01/01/1993 to 01/01/2020	109	107	86	21
3	Italy	01/01/1995 to 01/01/2020	101	99	79	20
4	Germany	01/01/1991 to 01/01/2020	117	115	92	23
5	Spain	01/01/1995 to 01/01/2020	101	99	79	20
6	France	01/01/1990 to 01/01/2020	121	119	95	24
7	Japan	01/01/1994 to 01/01/2020	105	103	82	21
8	India	01/04/2011 to 31/03/2020	36	34	28	06

predict GDP. The model has two hidden layers with 9 and 3 neurons and an output layer with one neuron. Initially, all the connecting weights and bias values are randomly selected between -0.5 and 0.5 . The choice of initial values affects the convergence speed and the final mean square error after the training phase. To make the training process unbiased, the initial weights are randomly selected. The final weights after the training process may have positive or negative values; therefore, the initial weights are chosen to be both negative and positive values. Further, to achieve unbiased selection, the weights are chosen from a uniform distribution with zero mean (unbiased) ranging between -0.5 and 0.5 . The total numbers of weights required between the input layer and first hidden layer, first hidden layer and second hidden layer, and second hidden layer and output layer are 5×9 , 9×3 , and 3×1 , respectively. Similarly, the numbers of biases are 9, 3, and 1 for the first hidden, second hidden, and output layers, respectively. The first data pattern with five values is fed to the model, and the input is weighted and passed through an activation function to produce an output at the first hidden layer. The same process is repeated for the second hidden layer and finally the output layer. The output is then compared with the corresponding target value to calculate the error. The BP rule is used to update the weights and bias values using Eqs. (6)–(11). In the same way, all input patterns are fed to the model sequentially, and the weights are updated until all input patterns are exhausted. This process completes one experiment and is repeated 50,000 times in 10 independent runs. The value of the learning parameter is 0.1. During every experiment, the mean squared error (MSE) is calculated and plotted to illustrate the convergence characteristics of the model. The error convergence plots during training are shown in Figs. 4(a)–(h), and a comparison of the actual and predicted values of the model during training is shown in Figs. 3(a)–(h). Once the error is minimized, the training process is stopped, and the final values of the weights and biases are saved for testing purposes.

(iv) Testing or validation of the model: Testing or validation of the model is performed using the 20% of the input or data patterns that are not used during training. The testing patterns are input into the trained model sequentially, and the output is obtained after weighting, adding, and passing through the activation function. Each of the outputs of the model is compared with the available target value to calculate the mean absolute percentage error (MAPE) using (12)

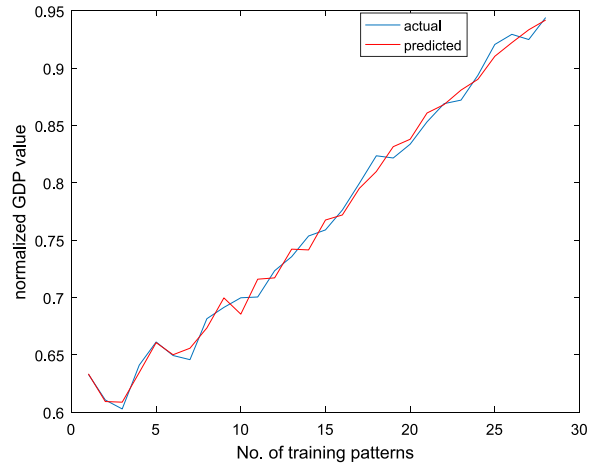
$$MAPE = \frac{1}{L} \sum_{l=1}^L \text{abs}((a(l) - p(l))/a(l)) \times 100 \quad (12)$$

where $a(l)$ = actual value of the l th testing pattern
 $p(l)$ = predicted value of the l th testing pattern

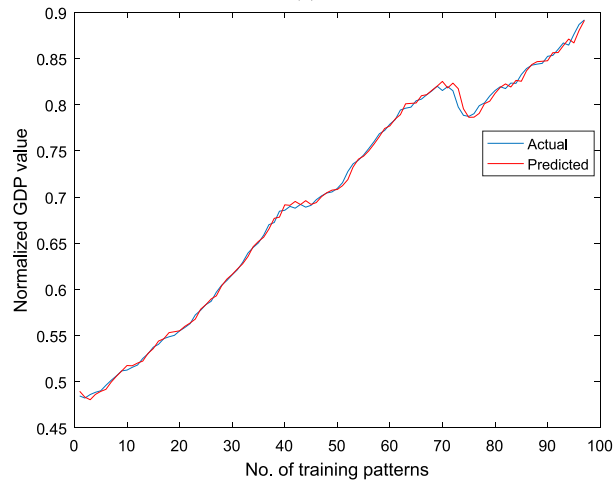
4. Simulation results

Figs. 3(a) to 3(h) present a comparison of the actual data and the forecasted estimates in the training process. The model has been trained accurately to capture the spikes present in the data.

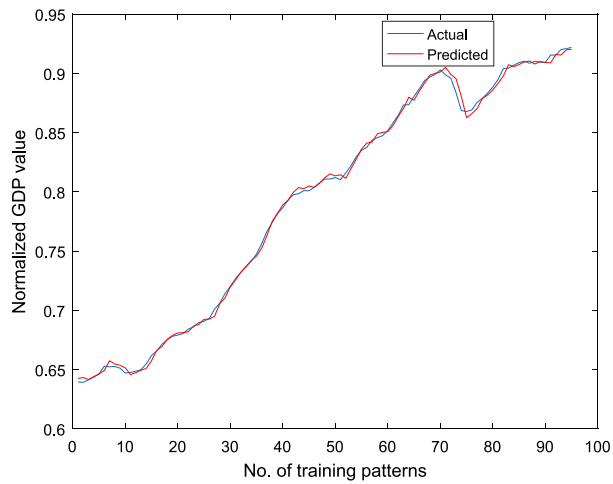
Figs. 4(a) to 4(h) show the MSE of the estimated GDP outputs obtained from the training of the model. The figures show that the MSE curve starts with a high value but declines with each iteration and finally becomes parallel to the X-axis, showing that it has reached its minimum. Additional iterations do not further reduce the MSE, and the weights obtained from the model are optimum.



3(a) India

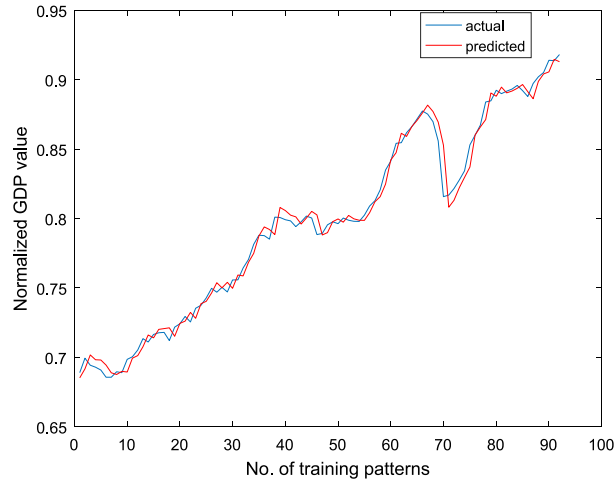


3(b) United States

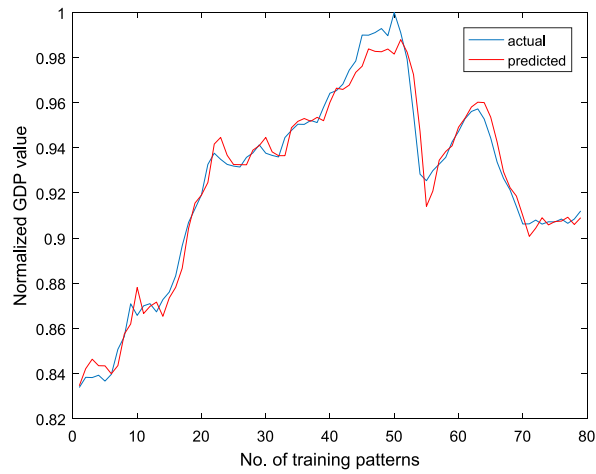


3(c) France

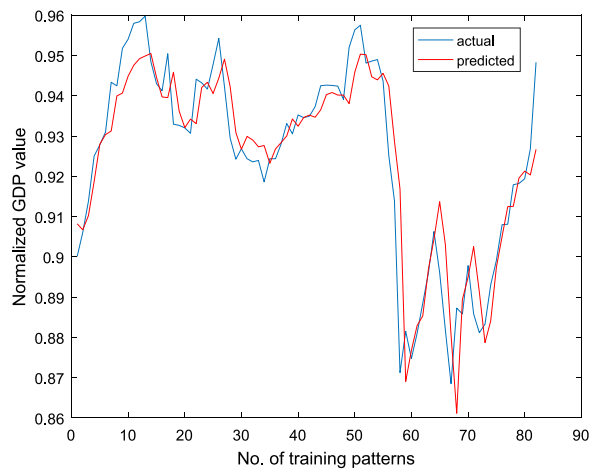
Fig. 3. Comparison of Actual and Predicted GDP during Training of the Model.



3(d) Germany

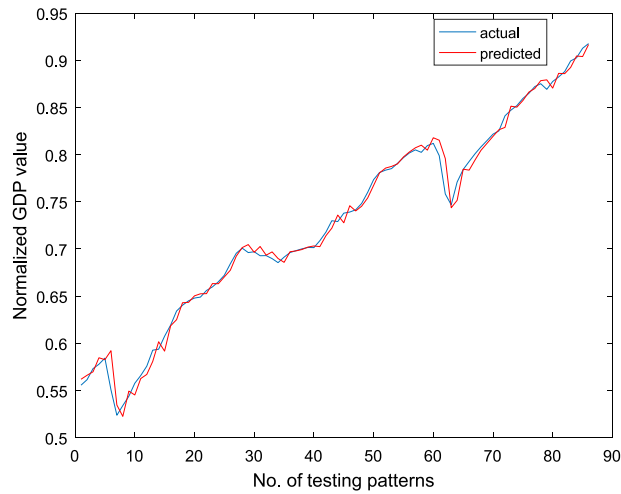


3(e) Italy

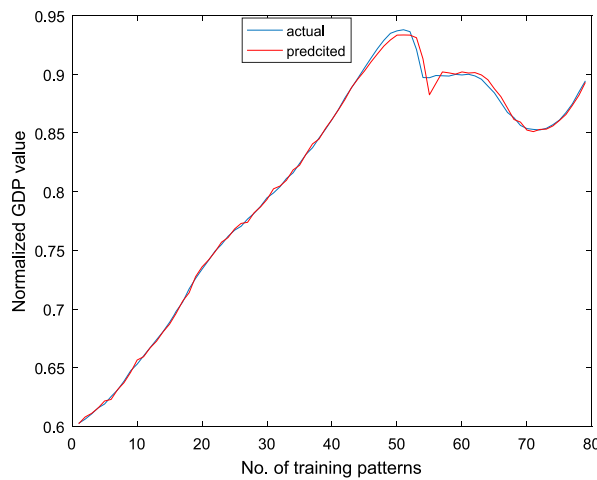


3(f) Japan

Fig. 3. (continued).



3(g) Mexico



3(h) Spain

Fig. 3. (continued).

Table 2 below shows the MAPE of the predicted GDP values during testing of the MLANN model for the 8 countries. The MAPE is an indicator of how close the predicted values are to the actual values. In all countries, the MAPE is 2% or less, which suggests that the ANN model developed to predict future values is well calibrated. The last column of Table 2 presents the forecasted GDP figures for the 8 countries considered in this study.

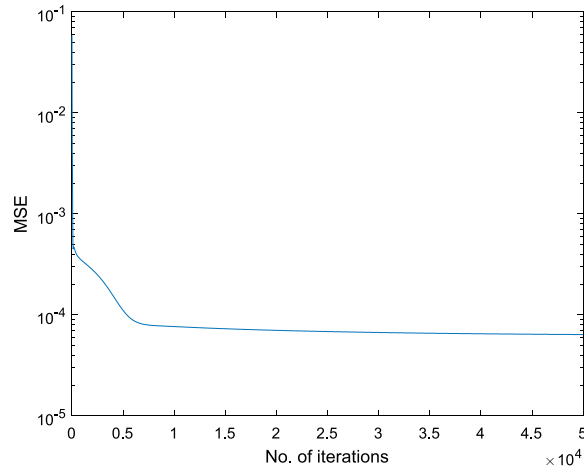
A comparison of the actual and predicted values of quarterly GDP for the countries during the testing of the model is provided in Appendices A and B.

5. Discussion of the results

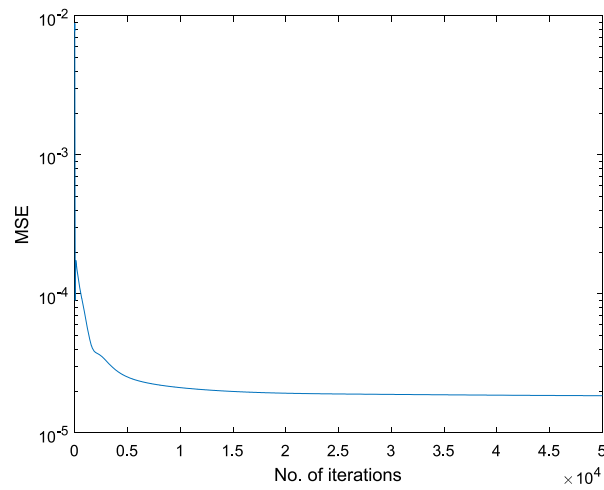
Table 3 presents the quarter-to-quarter growth rate for Q1 (January–March) and Q2 (April–June) for the 8 major economies. The Q1 growth rate is estimated using actual data, whereas the Q2 growth rate is based on the percentage change between the forecasted GDP figure of Q2 and the actual figure for Q1. Furthermore, the annual growth rate for all countries is given in the last column of Table 3. The quarter-to-quarter and annual growth rates are based on the following formula:

$$\text{Quarter GDP Growth } (G_q) = \frac{GDP_q - GDP_{q-1}}{GDP_{q-1}} \tag{13}$$

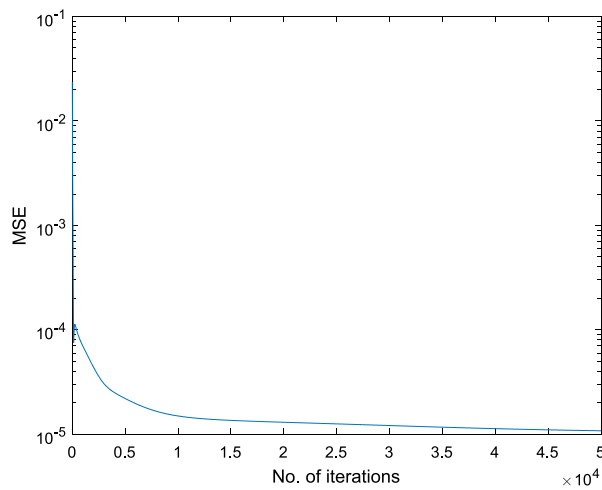
$$\text{GDP Annual } (G_a) = (1 + G_q)^4 - 1 \tag{14}$$



4(a) India

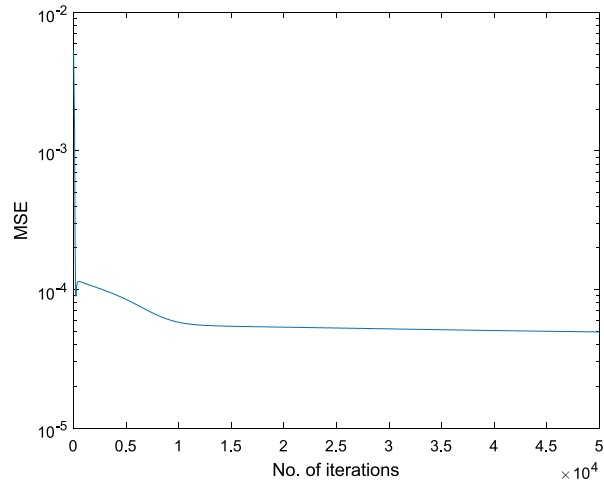


4(b) United States

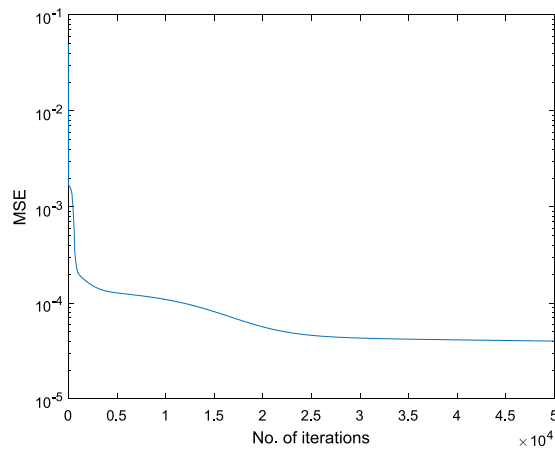


4(c) France

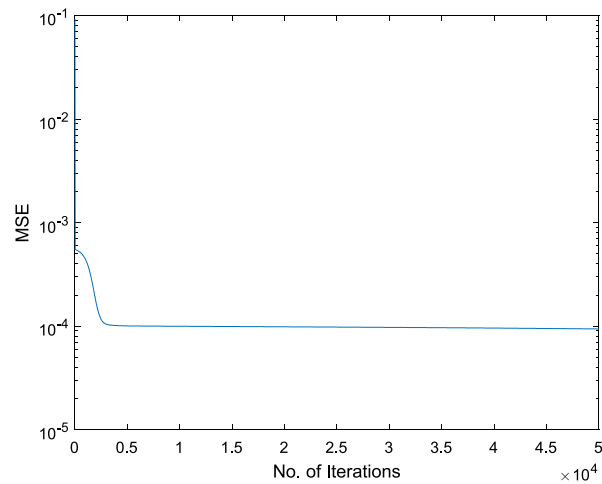
Fig. 4. MSE of Predicted GDP during Training of the Model.



4(d) Germany

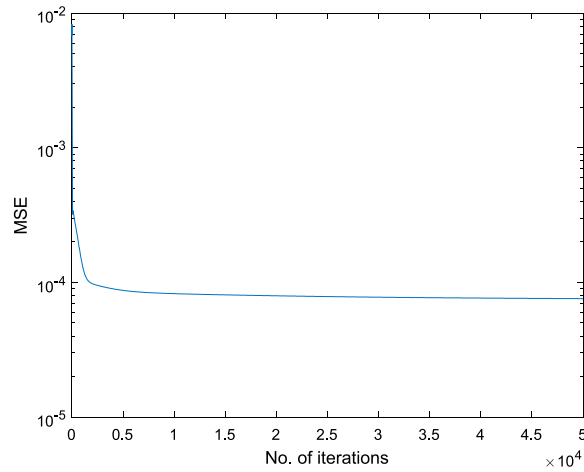


4(e) Italy

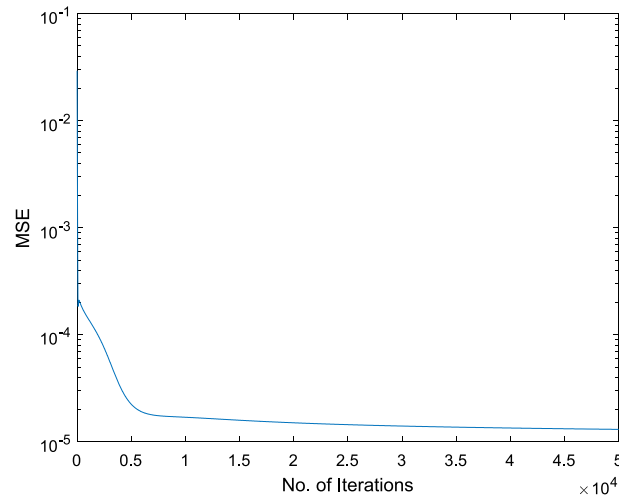


4(f) Japan

Fig. 4. (continued).



4(g) Mexico



4(h) Spain

Fig. 4. (continued).

Table 2
Mean absolute percentage error (MAPE) of GDP prediction during testing.

Country	MAPE during testing	Forecasted GDP value for April–June quarter (in local currency)
India	1.2503	33026
US	1.4458	18454
Germany	1.9925	724250
Japan	2.0429	533590
Italy	0.4479	373980
Spain	1.5360	274840
Mexico	1.9523	4450
France	1.5953	522350

Note: The forecasted GDPs of seven countries are represented in their respective currency – India in billion Rupee, United States in billion dollars, Germany, Spain, France and Italy in million Euro, Japan in billion Yen, Mexico in billion pesos.

The extreme severity of COVID-19 has prompted several governments to take necessary precautions based on the spread of infections. These measures include localized recommendations, national recommendations, and regional and

Table 3
Quarterly growth rates during COVID-19 shutdown.

Country	Quarterly growth (Jan–Mar, 2020) based on actual data	Quarterly growth (Apr–Jun, 2020) based on forecasted figure	Annual growth rate
United States	–1.29	–2.74	–10.53
Mexico	–1.60	–2.14	–8.30
France	–5.34	–1.78	–6.95
Spain	–5.24	–2.72	–10.45
Italy	–5.31	–2.40	–9.26
Germany	–2.22	–1.04	–4.11
Japan	–0.49	–2.42	–9.35
India	2.38	–2.78	–10.67

Note: Quarterly growth rate represents present quarter to previous quarter growth. Annual growth rate indicates the growth estimated at an annual rate, including all four quarters.

national lockdowns. As the disease containment measures became increasingly stringent by mid-March 2020, the disruption of supply chains and slowing of economic activities became drastic. However, a significant decline in worldwide economic growth was already noted even before COVID-19, mainly due to the trade war between China and the United States and the steep decline in consumer expenditure ([World Economic Outlook Update, 2019](#)).

[Table 3](#) shows that most economies shrank in the 1st quarter of 2020. The United States, the largest economy in the world, experienced a 1.3% decline in GDP growth in the January to April quarter when compared to the previous quarter. The US economy initiated regional restrictions on movement based on the severity of the pandemic in mid-March. However, the first quarter captured only 2 weeks of the lockdown, and the impact loomed large in the second quarter due to increased pandemic outbreaks and restrictions in different states of the country. Mexico experienced a situation similar to that of the US, where restrictions were initiated in mid-March and GDP growth fell by 1.6% in the first quarter of 2020.

While North American countries restricted movements to a certain extent to curb the spread of the disease, European countries, such as Italy, Spain, Germany, and France – the four largest Euro economies that had the highest number of infections – imposed a stricter form of lockdown. The strict lockdown measures caused economic contractions of 5.34%, 5.24%, and 5.31% in France, Spain, and Italy, respectively, in the first quarter of 2020. The largest economy of the Eurozone, Germany, experienced a 2.2% GDP contraction during the same time. Although flattening of the pandemic outbreak curve was achieved in European countries by the end of June, the second quarter of 2020 may entail a steep decline in GDP due to the complete shutdown of economic activities.

The greatest impact is observed in countries such as France and Spain, which have a strong dependence on tourism and the service industry compared to Germany. Asian countries showed modest growth from January to March. India's GDP grew at 2.38% in this quarter compared to the previous quarter. India imposed a nationwide lockdown on the 25th of March 2020. Hence, the January to March growth rate does not reflect the impact of the lockdown; rather, it reflects the already slowing economy even before the outbreak of COVID-19. Surprisingly, air pollution, as a byproduct of economic activity, has decreased during this period ([Kumar and Managi, 2020](#)). The Japanese economy contracted by 0.5% from January to March compared to the previous quarter, the second straight decline in economic growth for two consecutive quarters. Reduced exports amid a trade war between the US and China, followed by a slump in consumer expenditure, have had a severe impact on the Japanese economy.

The devastating spread of the pandemic and extension of restrictions among several countries has resulted in a steep decline in economic growth and led to recession ([IMF, 2020](#)). The decline in aggregate demand caused increased layoffs, leading to higher unemployment. The key policies by the central banks of different countries to lower the impact and induce liquidity are evident. The world economy was on the cliff, with significant events such as geopolitical tensions and trade wars, and the coronavirus pushed the economy over the edge ([World Bank, 2020](#); [United Nations, 2020](#)).

We forecast the GDP growth for the 2nd quarter of 2020 because of the policy implications. The forecasted GDP for the April to June quarter and the annualized GDP growth indicate a steep decline among the major economies of the world. The forecasted GDP growth rates in the United States and Mexico are –2.74% and –2.4%, respectively, for the 2nd quarter. The restrictions in these countries have disrupted the movement of several essential and nonessential goods. However, what is more concerning is the double-digit decline in annualized GDP growth, which stands at –10.53% and –8.3% in the US and Mexico, respectively. The decreased demand for petroleum products at home and export goods abroad might have caused such a substantial negative impact on the US economy. Furthermore, increased unemployment and severe disruption of service sector-related industries, such as hotels and airlines, may lead to a sharp decline in economic growth and personal consumption.

The disruption in the Eurozone is expected take quite a long time to recover, as most of the member countries are heavily dependent on tourism and services. The major economies of the Eurozone – France, Spain, Italy, and Germany – are predicted to contract by 1.78, 2.72, 2.4, and 1.04%, respectively. The unemployment stimulus package in the Euro area is smaller than that in the United States due to the widespread use of short-term work policies ([World Bank, 2020](#)). Among Asian countries, Japan prevented the spread of the pandemic by means of national recommendations. Although the

Table 4
Comparison of GDP annual growth forecasts by our model and other leading global agencies.

Country	Our model	IMF	World Bank	Morgan Stanley
United States	−10.53	−8	−7.9	−5.8
Mexico	−8.30	−10.5	−8.7	
France	−6.95	−12.5		
Spain	−10.45	−12.8		
Germany	−4.11	−7.8	−10.1	−9.6
Italy	−9.26	−12.8		
Japan	−9.35	−5.8	−6.8	
India	−10.67	−4.5	−9	−1.7

Note: The GDP growth rates estimated by various financial agencies were taken from different reports for comparison purposes.

Japanese economy was opened for trade, a strong contraction in GDP is observed in the second quarter, which follows the decline experienced from October to December 2019. The spillover effect of the trade war and decline in global aggregate demand may continue, which would further hinder the growth of the Japanese economy. While predicting Japan's GDP in the training dataset (see Fig. 3f), a significant gap was observed between the actual and predicted values for some of the previous quarters. One explanation of this gap is that the predictive accuracy of a model is affected when an economy faces frequent uncertain events. This is what occurred for the Japanese economy, as it has been faced with several uncertain policy decisions resulting from an unsustainable fiscal trajectory, constraints on monetary policy, uncertainty around world trade policies, and weak growth (Arbatli et al., 2019). A difference between the actual and predicted values of GDP is also observed for Italy (see Fig. 3e). This prediction error may be explained by the prolonged period of policy uncertainty in both countries.

In India, a nationwide lockdown was initiated in the last week of March and was continued in the subsequent months of the next quarter with localized lockdown and nationwide recommendations. The forecasted growth rate of GDP in India is negative, i.e., -2.78% , and the annualized growth rate for 2020–21 is predicted to be -10.67% . Towards the end of the April–June quarter, the pandemic was raging through rural areas of the country as the migrant reverse exodus occurred, in which millions of migrant workers returned to their native rural homes from the cities. At the time of writing this paper, the pandemic had affected 1.7 million lives in India, and several states were in and out of lockdown. Therefore, a significant contraction in GDP growth in India, as forecasted by us, is not surprising. While the fiscal stimulus provided by different central banks would reduce the contraction in the 2nd quarter, the overall economic outlook in most of the countries in 2020 looks bleak.

In Table 4, the forecasts from the ANN model are compared with the forecasts from leading global agencies to enhance the understanding of the world economic outlook in 2020. Morgan Stanley's outlook for 2020 is optimistic, with a V-shaped recovery, and their forecasts show a modest decline in GDP for the 8 countries we considered. Their optimism is based on the slender recovery in economic activities observed in May. The projections by the World Bank and IMF are closer to our forecasts. The forecasted decline in GDP growth for all these countries warrants a strong response from governments to prevent economies from slipping into recession.

Although past economic shocks have contracted the world economy, the global financial crisis of 2008 was the greatest slump since the great depression in 1930. Low-interest rate policies, overleveraging, and unsustainable fiscal and monetary policies led to a global financial crisis, causing a credit crunch and unemployment, which pushed world economies into a deep recession. World GDP growth slowed from 5% in 2007 to 3.75% in 2008 and 2% in 2009. However, with remedial measures, such as quantitative easing, financial reforms by the governments and central banks, signs of recovery were evident in 2010. Other past pandemic outbreaks had negative effects on economies, but they were limited to a few countries. The 1918 Spanish influenza had a significant impact on major economies. The projected reduction in quarterly GDP was 2.6% for the United States (Dixon et al., 2010), and strong evidence of a negative effect on capital returns was found in Sweden (Karlsson et al., 2014). The short-term impact of Ebola on most African economies was negative, as GDP growth fell by 2.1% in Guinea, 3.4% in Liberia, and 3.3% in Sierra Leone (World Bank, 2014). The specific countries facing pandemics in the past had a small impact on the world economy, while the COVID-19 pandemic has had a widespread negative impact on global economic growth and trade. The ongoing trade war and uncertain worldwide events, followed by the pandemic outbreak, have led to an unprecedented economic crisis. This uncertain nature of the economic impact of the pandemic has resulted in a gap between the actual and forecasted quarterly GDP growth during the COVID-19 shutdown period. The quarter-to-quarter fluctuations in GDP are the reason for the forecasting error during the shutdown period. The current model, which has accurately predicted the GDP during the training period, needs improvement to capture the large fluctuations.

6. Conclusion

The main contribution of this paper is the development of an ANN model to forecast GDP one quarter ahead for eight major economies. This model captures the nonlinearities present in the quarterly time-series data and provides accurate

Table A.1
Actual and predicted value obtained during testing pattern.

United States		Germany		Japan		Italy	
1	2	1	2	1	2	1	2
1.714	1.713	6.880	6.843	5.293	5.180	3.885	3.873
1.728	1.720	6.939	6.869	5.317	5.262	3.894	3.896
1.741	1.732	6.925	6.927	5.329	5.284	3.913	3.905
1.746	1.743	6.968	6.902	5.316	5.279	3.926	3.928
1.747	1.747	7.000	6.938	5.353	5.270	3.934	3.941
1.756	1.747	7.030	6.967	5.344	5.288	3.955	3.945
1.764	1.755	7.086	6.987	5.353	5.289	3.967	3.971
1.774	1.762	7.127	7.031	5.369	5.288	3.990	3.983
1.782	1.770	7.142	7.058	5.407	5.298	4.004	4.008
1.793	1.777	7.167	7.060	5.432	5.318	4.019	4.022
1.802	1.784	7.253	7.075	5.491	5.331	4.041	4.035
1.816	1.791	7.293	7.139	5.510	5.355	4.041	4.057
1.832	1.801	7.356	7.160	5.475	5.366	4.041	4.051
1.844	1.812	7.410	7.191	5.490	5.348	4.037	4.047
1.860	1.819	7.420	7.218	5.455	5.350	4.044	4.043
1.873	1.829	7.448	7.215	5.463	5.339	4.052	4.051
1.878	1.836	7.441	7.227	5.525	5.338	4.055	4.062
1.893	1.838	7.456	7.220	5.559	5.367	4.056	4.063
1.902	1.846	7.491	7.227	5.581	5.384	4.047	4.062
1.912	1.850	7.473	7.247	5.495	5.390	3.832	4.049
1.922	1.854	7.493	7.233	5.469	5.357		
1.898	1.859						

Note: The table includes the total number of testing pattern and obtained (1) Actual values and (2) Predicted Values for different countries.

predictions. These countries have experienced substantial negative health impacts from the ongoing COVID-19 pandemic. The infection and fatality rates have been alarming, and governments have implemented various forms of lockdown to contain the disease. As a result, their economies have been disrupted, resulting in the shutting down of many industries and rising unemployment rates. In such a situation, a clear picture of what lays ahead in terms of economic outlook will help policymakers take necessary steps. For example, opinion has been divided in the US Congress as to how much relief is sufficient to restart the economy. The ANN model developed in our paper accurately predicted the GDP figures, as the MAPE is less than 2% in each of the country cases.

The findings show that the April to June quarters of 2020 will see a significant decline in economic growth in all eight countries. The annualized GDP growth shows an even larger impact, as most countries will experience double-digit negative economic growth. Such a scenario is expected, though it requires strong corrective actions by central banks and governments. The US government has announced a rescue package of \$2.2 trillion ([The Hindu, 2020](#)), which is the largest rescue package in recent decades. Joseph Stiglitz, the Nobel laureate in Economic Sciences, in an interview with the British tabloid *The Independent* has opined that an even larger rescue package to the tune of \$6 trillion (a third of the US GDP) may be required. Similarly, the Indian government declared a \$260 billion coronavirus rescue package ([The New York Times, 2020](#)). This rescue package is supposed to support small- and medium-scale industries and the agriculture sector, reinstate migrant laborers, and support ailing banks and financial institutions. Furthermore, most central banks have injected liquidity into the economy by reducing interest rates.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix A

See [Table A.1](#).

Appendix B

See [Table B.1](#).

Table B.1
Actual and predicted value obtained during testing pattern.

France		Mexico		Spain	
1	2	1	2	1	2
5.184	5.175	4.265	4.261	2.667	2.663
5.208	5.178	4.309	4.268	2.691	2.689
5.215	5.201	4.362	4.313	2.717	2.708
5.242	5.205	4.361	4.350	2.736	2.729
5.239	5.227	4.387	4.336	2.747	2.742
5.256	5.221	4.407	4.370	2.771	2.746
5.262	5.235	4.455	4.377	2.784	2.768
5.300	5.239	4.506	4.415	2.805	2.776
5.287	5.271	4.527	4.441	2.834	2.791
5.299	5.256	4.542	4.449	2.849	2.814
5.329	5.264	4.528	4.459	2.869	2.821
5.374	5.290	4.581	4.448	2.883	2.832
5.411	5.323	4.642	4.488	2.899	2.840
5.450	5.346	4.630	4.509	2.913	2.848
5.495	5.368	4.645	4.494	2.930	2.856
5.504	5.393	4.648	4.511	2.946	2.864
5.515	5.394	4.642	4.508	2.957	2.872
5.534	5.398	4.637	4.506	2.969	2.877
5.569	5.408	4.633	4.505	2.982	2.882
5.598	5.427	4.627	4.503	2.825	2.887
5.612	5.440	4.553	4.500		
5.624	5.445				
5.619	5.449				
5.318	5.445				

Note: The table includes the total number of testing pattern and obtained (1) Actual values and (2) Predicted Values for different countries.

References

- Acemoglu, D., Chernozhukov, V., Werning, I., Whinston, M., 2020. A multi-risk SIR model with optimally targeted lockdown. *Natl. Bur. Econ. Res.* <http://dx.doi.org/10.3386/w27102>.
- Ali Choudhary, M., Haider, A., 2012. Neural network models for inflation forecasting: An appraisal. *Appl. Econ.* 44, 2631–2635. <http://dx.doi.org/10.1080/00036846.2011.566190>.
- Arbatli, E.C., Davis, A., Ito, S.J., Miale, N., 2019. Policy Uncertainty In Japan. NBER WORKING PAPER SERIES, Working Paper 23411. <http://www.nber.org/papers/w23411>.
- Balcilar, M., Gupta, R., Majumdar, A., Miller, S.M., 2015. Was the recent downturn in US real GDP predictable? *Appl. Econ.* 47, 2985–3007. <http://dx.doi.org/10.1080/00036846.2015.1011317>.
- Bañbura, M., Giannone, D., Reichlin, L., 2010. Large Bayesian vector auto regressions. *J. Appl. Econom.* 25, 71–92. <http://dx.doi.org/10.1002/jae.1137>.
- Box, G., Jenkins, G., Reinsel, G., Ljung, G., 2015. *Time series analysis: forecasting and control*.
- Chuku, C., Simpasa, A., Oduor, J., 2019. Intelligent forecasting of economic growth for developing economies. *Int. Econ.* 159, 74–93. <http://dx.doi.org/10.1016/j.inteco.2019.06.001>.
- Dixon, P.B., Lee, B., Muehlenbeck, T., Rimmer, M.T., Rose, A.Z., Verikios, G., 2010. Effects on the U.S. of an H1N1 epidemic: analysis with a quarterly CGE model. Centre of Policy Studies/IMPACT Centre Working Papers g-202, Victoria University, Centre of Policy Studies/IMPACT Centre.
- FRED | St. Louis Fed, 2020. Economic research division, Federal Reserve Bank of St. Louis. <https://fred.stlouisfed.org>.
- Guégan, D., Rakotomaroahy, P., 2010. Alternative methods for forecasting GDP. *Int. Symp. Econ. Theory Econ.* 20, 161–185. [http://dx.doi.org/10.1108/S1571-0386\(2010\)0000020013](http://dx.doi.org/10.1108/S1571-0386(2010)0000020013).
- Härdle, W., Müller, M., Sperlich, S., Werwatz, A., 2012. Nonparametric and semiparametric models. http://dx.doi.org/10.1007/978-3-642-17146-8_1.
- Haykin, Simon, 2009. *Neural networks and learning machines (3rd edition)* by Simon Haykin. Neural Netw..
- Hlaváček, H.H., Koňák, M., Koňák, K., Michael, 2005. *The Application of Structured Feedforward Neural Networks To the Modelling of Daily Series of Currency in Circulation*. Springer.
- IMF, 2020. World Economic Outlook Update: A Crisis Like No Other, An Uncertain Recovery, World Economic Outlook Update.
- International Monetary Fund, 2020. World economic outlook, january 2020, tentative stabilization, sluggish recovery? *World Econ. Outlook* 1–10.
- Jahn, M., 2020. Artificial neural network regression models in a panel setting: Predicting economic growth. *Econ. Model.* 91, 148–154. <http://dx.doi.org/10.1016/j.econmod.2020.06.008>.
- Jena, P.R., Majhi, R., Majhi, B., 2015. Development and performance evaluation of a novel knowledge guided artificial neural network (KGANN) model for exchange rate prediction. *J. King Saud. Univ. - Comput. Inf. Sci.* 27, 450–457. <http://dx.doi.org/10.1016/j.jksuci.2015.01.002>.
- Karlsson, M., Nilsson, T., Pichler, S., 2014. The impact of the 1918 spanish flu epidemic on economic performance in Sweden an investigation into the consequences of an extraordinary mortality shock. *J. Health Econ.* 36, 1–19.
- Kock, A.B., Teräsvirta, T., 2014. Forecasting performances of three automated modelling techniques during the economic crisis 2007–2009. *Int. J. Forecast.* 30, 616–631. <http://dx.doi.org/10.1016/j.ijforecast.2013.01.003>.
- Kumar, S., Managi, S., 2020. Does stringency of lockdown affect air quality? Evidence from Indian cities. *Econ. Disasters Clim. Change* 4, 481–502.
- Loermann, J., Maas, B., 2019. Munich Personal RePEc Archive Nowcasting US GDP with artificial neural networks.
- McNelis, P.D., 2005. Neural networks in finance: gaining predictive edge in the market. <http://dx.doi.org/10.1017/CBO9781107415324.004>.
- McNelis, P.D., McAdam, P., 2004. *Forecasting inflation with thick models and neural networks*.
- Nakamura, H., Managi, S., 2020. Airport risk of importation and exportation of the COVID-19 pandemic. *Transp. Policy* 96, 40–47. <http://dx.doi.org/10.1016/j.tranpol.2020.06.018>.

- Ramos-Pérez, E., Alonso-González, P.J., Núñez Velázquez, J.J., 2019. Forecasting volatility with a stacked model based on a hybridized artificial neural network. *Expert Syst. Appl.* 129, 1–9. <http://dx.doi.org/10.1016/j.eswa.2019.03.046>.
- RBI, 2020a. Reserve bank of India. Monetary Policy Report April 2020.
- RBI, 2020b. Reserve bank of India. In: *HandBook of Statistics on Indian Economy*.
- Saman, C., 2011. Scenarios of the Romanian Gdp evolution with neural models 1. *Romanian J. Econ. Forecast.* 4.
- Shi, P., Chen, Z., Xie, G., 2006. Using artificial neural network trained with genetic algorithm to model GDP prediction.
- Sivanandam, S.N., Deepa, S.N., 2013. *Principles of Soft Computing*, second ed. Wiley.
- Szafranek, K., 2019. Bagged neural networks for forecasting polish (low) inflation. *Int. J. Forecast.* 35, 1042–1059. <http://dx.doi.org/10.1016/j.ijforecast.2019.04.007>.
- Teräsvirta, T., 2005. *A Service of Zbw Standard-Nutzungsbedingungen: Forecasting Economic Variables with Nonlinear Models*. Elsevier.
- Teräsvirta, T., 2006. Chapter 8 forecasting economic variables with nonlinear models. *Handb. Econ. Forecast.* 1, 413–457. [http://dx.doi.org/10.1016/S1574-0706\(05\)01008-6](http://dx.doi.org/10.1016/S1574-0706(05)01008-6).
- The Hindu, 2020. U.S. Senate unanimously passes \$2.2 trillion coronavirus rescue package. Accessed at <https://www.thehindu.com/news/international/us-senate-unanimously-passes-22-trillion-coronavirus-rescue-package/article31169709.ece>, on 1st August, 2020. (Accessed 8 August 20).
- The New York Times, 2020. Modi announces \$260 billion coronavirus rescue package for India. Accessed at <https://www.nytimes.com/2020/05/12/world/asia/virus-india-rescue-package.html>. (Accessed 1 August 2020).
- Tkacz, G., 2001a. Neural network forecasting of Canadian GDP growth. *Int. J. Forecast.*
- Tkacz, G., 2001b. Neural network forecasting of Canadian GDP growth. *Int. J. Forecast.* 17, 57–69. [http://dx.doi.org/10.1016/S0169-2070\(00\)00063-7](http://dx.doi.org/10.1016/S0169-2070(00)00063-7).
- Torres, D.G., Qiu, H., 2018. Applying Recurrent Neural Networks for Multivariate Time Series Forecasting of Volatile Financial Data. WHO.
- United Nations, New York, 2020. *World Economic Situation and Prospects*.
- WHO, 2020. *Monitoring Health for the SDGs*. WHO.
- World Bank, 2020. *World Bank Report. Global Economic Prospects*.
- World Economic Outlook Update, 2019. *A weakening global expansion*. (Accessed 8 August 20).
- Yoo, S., Managi, S., 2020. Global mortality benefits of COVID-19 action. *Technol. Forecast. Soc. Change* 160, 120231. <http://dx.doi.org/10.1016/j.techfore.2020.120231>.



Real-Time Identification of Fuzzy PID-Controlled Maglev System using TLBO-Based Functional Link Artificial Neural Network

Amit Kumar Sahoo¹ · Sudhansu Kumar Mishra² · Babita Majhi³ · Ganapati Panda⁴ · Suresh Chandra Satapathy⁵ 

Received: 30 May 2020 / Accepted: 30 December 2020
© King Fahd University of Petroleum & Minerals 2021

Abstract

In this paper, the teaching–learning-based optimization-based functional link artificial neural network (FLANN) has been proposed for the real-time identification of Maglev system. This proposed approach has been compared with some of the other state-of-the-art approaches, such as multilayer perceptron–backpropagation, FLANN least mean square, FLANN particle swarm optimization and FLANN black widow optimization. Further, the real-time Maglev system and the identified model are controlled by the Fuzzy PID controller in a closed loop system with proper choice of the controller parameters. The efficacy of the identified model is investigated by comparing the response of both the real-time and identified Fuzzy PID-controlled Maglev system. To validate the dominance of the proposed model, three nonparametric statistical tests, i.e., the sign test, Wilcoxon signed-rank test and Friedman test, are also performed.

Keywords System identification · Maglev system · FLANN · TLBO · Fuzzy PID

1 Introduction

In the recent past, many articles have been published on the identification of a complex system, owing to its widespread use in various areas. System identification means the estimation of the parameters of a plant or matching the output

responses of the model with that of the physical system. System identification is intended to find the deep understanding of the cause–effect relationships [1–4]. The nature of the system is categorized by different characteristics, such as its electrical, physical and chemical properties. However, it is very difficult to understand and model such characteristics of the plant. Thus, identification is a big challenge in several fields like control engineering [5, 6], power system engineering [7], renewable [8], etc. Accurate and quick identification is a difficult task for real-world plants which is mainly due to its nonlinear and dynamic nature. Many researchers have applied various forms of the artificial neural network (ANN) like multilayer perceptron (MLP) [9], functional link artificial neural network (FLANN) [10, 11], radial basis function (RBF) [12, 13], etc., for the identification purpose. By using multilayer perceptron (MLP) networks, Narendra and Parthasarathy have reported various identification techniques for a low complexity dynamic system [14]. However, the MLP network has multiple layers, which make it computationally expensive for the identification of any complex system. The FLANN model which is introduced by Pao et al. [15] is a single layer neural network without any hidden layer. The FLANN input is functionally expanded with different expansion techniques like power series, trigonometric, Chebyshev expansion, etc. This model is having lower computational complexity with a fast rate of convergence. The FLANN has

✉ Suresh Chandra Satapathy
suresh.satapathyfcs@kiit.ac.in

Amit Kumar Sahoo
amitkumar2687@gmail.com

Sudhansu Kumar Mishra
sudhansumishra@bitmesra.ac.in

Babita Majhi
babita.majhi@gmail.com

Ganapati Panda
ganapati.panda@gmail.com

¹ Department of EEE, Centurion University of Technology and Management, Bhubaneswar, Odisha, India

² Department of EEE, Birla Institute of Technology, Mesra, Ranchi, India

³ G. G. Vishwavidyalaya (A Central University), Bilaspur, CG 495009, India

⁴ C. V. Raman Global University, Bhubaneswar, India

⁵ KIIT University, Bhubaneswar, India



been used for pattern classification [16], prediction [17] and many other challenging tasks with faster convergence and lesser complexity compared to the MLP.

In the training phase of an ANN, all the weights are iteratively updated, and they reached the optimal value. The methods for updating the weights of a neural network could be based on the derivative or free from the derivative. Some of the examples of the derivative/gradient based are least mean squares (LMS) [6], backpropagation (BP) [2], recursive least squares (RLS) [18], etc. Similarly, examples of the second category include bio-inspired or evolutionary computing or computational intelligence-based approach. In most of the applications, the gradient-based approach provides inferior solutions due to the inherent limitations, such as trapping at local optimum points and incapability of finding derivatives of the discontinuous function.

To eliminate the above shortcomings, derivative-free algorithms, such as the genetic algorithm (GA) [19], particle swarm optimization (PSO) [20–22], and black widow optimization (BWO) [23], have been applied by different researchers to train the model. Kumar et al. [24] have introduced a metaheuristic-based socio-evolution and learning optimization algorithm (SELO) inspired by the social learning behavior of humans. The performance of the SELO is evaluated using 50 benchmark problems and compared with the other competitive algorithms. The results show the performance of the SELO is better than the others. Gholizadeh et al. [25] have introduced a metaheuristic algorithm, i.e., improved fireworks algorithm (IFWA) used for a discrete structural optimizations problems of steel trusses and frames. The optimization results demonstrate that the IFWA has highly competitive and superior over the standard FWA algorithm in terms of the convergence rate and statistical analysis. Gholizadeh et al. [26] have proposed a metaheuristic algorithm, center of mass optimization (CMO) to deal with performance-based discrete topology optimization (PBDTO) problem. PBDTO process is implemented for four multi-story steel braced frames by CMO. The authors have concluded that the CMO-based PBDTO formulation is an efficient technique for the seismic discrete topology optimization. Gholizadeh et al. [27] have proposed a new and efficient metaheuristic algorithm Newton metaheuristic algorithm (NMA) for optimization of steel moment frames. The NMA is a population-based framework which uses Newton gradient-based method. Here, the authors investigate the effectiveness of the proposed algorithm by considering two benchmark discrete trusses optimization problems. The performance of the proposed algorithm is analyzed on the basis of statistical parametric and nonparametric test and found to be superior over other competitive algorithms. Hayyolalam et al. [23] have proposed a novel black widow optimization algorithm (BWO), which is inspired by mating behavior of black widow spiders. The efficacy of the BWO

algorithm is determined by taking 51 different benchmark functions. From obtained results, it is confirmed that the BWO has better performance and superiority as compared to other algorithms. All these optimization techniques may be implemented to update the weights of the neural network and applied for identification of any system.

However, selecting the proper controlling parameters of these derivative-free bio-inspired algorithms is still a challenging task because of the presence of many controlling parameters. Due to these controlling parameters, the weight updation of neural network model is complex, computationally expensive and time consuming. Hence, there is a need to explore other bio-inspired algorithms with less number of controlling parameters. Rao et al. [28] recently came up with the TLBO optimization technique to circumvent the above shortcomings, which uses the teaching and learning methodology of the teacher and the student in a classroom. They highlighted the merits of TLBO that it does not depend on any controlling parameters, and only need the algorithm specific parameters, such as number of populations, iterations and stopping criteria. They have stressed on the fact that the TLBO eliminates the intricacy of the optimum selection and optimization of controlling parameters, which is usually necessary in other bio-inspired techniques. Naik et al. [29] have concluded that the performance of higher order neural networks is sensitive to weight initialization and relies on a kind of adopted learning algorithm. They have implemented TLBO for the training of ANN's, and applied it successfully for the classification problem. In this manuscript, we have implemented TLBO for optimizing weights of a variant of ANN, i.e., FLANN for identification of Maglev plant.

In this paper, MLP-BP, FLANN-LMS, FLANN-PSO, FLANN-TLBO and FLANN-BWO have been implemented for the identification of the Maglev system. The comparative analysis of performance among all these approaches is carried out by considering the mean squares error and the computational time. Here, a Fuzzy PID controller is also implemented to control the identified model, and then, the response is compared with that of the Fuzzy PID-controlled actual Maglev system.

The organization of the paper is as follows: Introduction and the recent work on identification are presented in Sect. 1. Section 2 presented and illustrated the construction and principle of the Maglev plant as shown in Fig. 1. Discussion of related work is presented in Sect. 3, and Sect. 4 highlighted the prerequisites of the research work. Section 5 deals with the proposed TLBO-based FLANN model for identification of the Maglev plant. In Sect. 6, design of controller based on the Fuzzy PID is discussed. In Sect. 7, the simulation study, validation and nonparametric statistical test of proposed model and the results of top-notch models are presented and compared. Section 8 presents the contribution of the manuscript, and the scope of future research work is outlined.

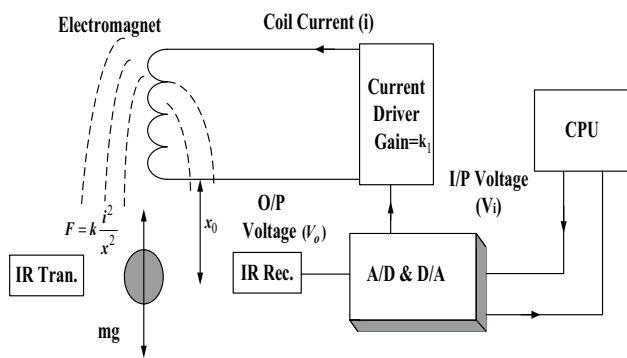


Fig. 1 Schematic diagram of the Maglev system



Fig. 2 The Maglev laboratory setup

2 The Magnetic Levitation Plant

The laboratory setup of the Maglev system from Feedback Instruments Ltd., Model No. 33-210 is shown in Fig. 2, and it has a wide range of applications like magnetically balanced bearings, vibration damping and transportation systems (i.e., very popularly known as Maglev trains) [30–33]. Basically, it works on the Maglev principle and has two parts: (i) the Maglev plant, and (ii) the digital computer where the controlling action takes place. The Maglev system comprises of different integrated components, like the electromagnet, ferromagnetic ball, IR sensor and a current driver circuit. A digital computer provides an immense platform for the effective design of various controllers, which can be implemented using MATLAB and Simulink for real-time applications. The whole setup accumulates both mechanical and electrical units with I/O interface systems.

The Maglev plant parameters are given in Table 1, and its transfer function is as follows [34–36]:

$$G_p(s) = \frac{\Delta V_o}{\Delta V_i} = \frac{-3518.85}{s^2 - 2180} \quad (1)$$

Table 1 The physical parameters of the Maglev system

Name of the parameter	Symbol	Value
Mass of steel ball	m	0.02 kg
Control voltage to current gain (constant)	k_1	1.05 A/V
Sensor gain (constant), offset	k_2, η	143.48 V/m, -2.8 V
Input voltage to the controller	V_i	+ 5 V and - 5 V
Output voltage of sensor	V_o	+ 1.25 V to - 3.75 V
Equilibrium position of steel ball	x_0	0.009 m
Current at equilibrium position	i_0	0.8 A
Gravitational constant	g	9.81m/s ²

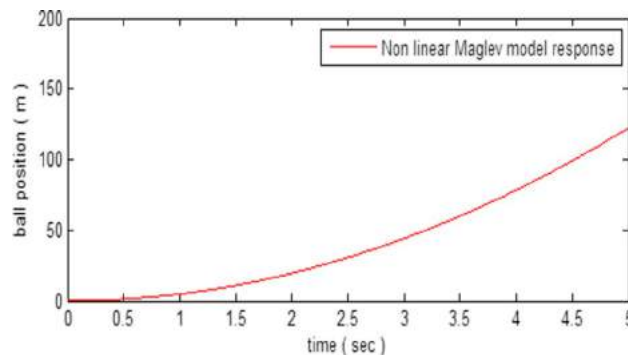


Fig. 3 Nonlinear response of Maglev system

where $G_p(s)$ represents the Maglev plant (Feedback Instruments Ltd., Model No. 33-210) transfer function, V_o is the output voltage of the sensor and V_i is the input voltage to the controller. From Eq. (1) and Fig. 3, it is found that the behavior of the Maglev system is highly nonlinear and unstable in nature. Therefore, it is challenging task to get the improved identified model of the Maglev plant.

3 Related Work

Artificial neural network (ANN) plays an important role in the identification of a nonlinear system [37, 38]. The neural network (NN) can performed nonlinear mapping between the input and output, as it has interconnection between the different layers. The neural network can be classified on the basis of its input, hidden and output layers. From the structural point of view, an ANN may be a single layer or multilayers. In a multilayer perceptron (MLP), there may be one or many hidden layers in between the input and output layers [39]. However, in a single layer structure, no hidden layer is present. Each neuron is connected from one layer to next layer of other neuron.

The learning of any neural network is a process where the weights are updated iteratively. These learning

processes may be of derivative based or derivative free. Some of the standard conventional gradient derivative-based approaches are LMS, RLS and BP algorithms. These have been applied by different researchers to train various neural networks and other adaptive models. Similarly, different derivative-free/evolutionary/bio-inspired learning algorithms, such as the GA, PSO, ant colony optimization (ACO), cat swarm optimization (CSO) and TLBO algorithms, are also used to train different neural network models.

Derivative-based algorithms are usually based on the gradient descent search algorithm and mathematically derived by utilizing the derivative of error. Least mean squares (LMS) are a stochastic gradient method or a simple derivative-based algorithm [40]. It is very popular, and widely used for its simple structure and ease of implementation to minimize the error. It is suitable for single layer ANN models for updating the weights. Backpropagation (BP) algorithm is derivative-based algorithm, which is suitable for multilayer ANN models [41]. The gradient-based optimization techniques fail to solve optimizing functions having discontinuities. These techniques may get trapped at local optimum points while solving functions having multiple optimal (maxima/minima) points. To overcome these bottlenecks of the traditional derivative-based approaches, different heuristic algorithms have been implemented by researchers. The PSO, which is based on the principle of the movement of a flock of birds that collectively search for food is a heuristic algorithm that has better convergence characteristics even for non-convex and discontinuous functions [20, 42]. This algorithm has a better exploration capability as the best among the swarm is followed by all the individuals along with their own best positions. The algorithm has provision for both local and global search techniques. The teaching–learning-based optimization (TLBO) has no control parameters. It undergoes a two phase search; the teacher phase performs a global search for better exploration, while the learner phase carries out for local search for better exploitation [43–45]. Also, this algorithm being dependent only on algorithm specific parameters, and without having controlling parameters is expected to have a better convergence characteristic is discussed in details in Sect. 4. The black widow optimization is a type of evolutionary-based optimization technique that imitates the strange mating behavior of the black widow spiders [23]. It is one of the latest techniques in the evolutionary-based optimization family. It delivers fast convergence speed, and avoids local optima problem. These techniques update the weights in three stages, i.e., procreate, cannibalism (sexual cannibalism and sibling cannibalism) and mutation.

4 Prerequisites

In this paper, a MLP and a special variant of ANN, i.e., FLANN is implemented for the identification of the Maglev system. FLANN is a type of single layer NN in which, the input data is allowed to pass through a functional expansion block, and hence, the input is functionally expanded with different expansion techniques. The power series expansion, trigonometric expansion and Chebyshev expansion are some of the mostly used expansion techniques. The Chebyshev functional expansion is found to be better for many engineering applications, and hence, it is considered for the expansion of FLANN inputs for the identification of Maglev system in this article. The Chebyshev expansion of input x_l , can be written as [41, 46, 47],

$$\begin{aligned}
 T_0(x_l) &= 1 \quad \text{for } l = 0 \\
 T_1(x_l) &= x_l \quad \text{for } l = 1 \\
 T_2(x_l) &= 2x_l^2 - 1 \quad \text{for } l = 2 \\
 T_{l+1}(x_l) &= 2x_l T_l(x_l) - T_{l-1}(x_l) \quad \text{for } l > 2
 \end{aligned}
 \tag{2}$$

The higher order polynomials are expanded as per usual practice. The output of the functional expansion block is multiplied with a set of weights. The basic structure of FLANN model that is trained by any adaptive algorithm is depicted in Fig. 4.

The lower computational complexity of the FLANN model due to its simple single layer structure, and simple learning algorithms, makes it computationally cheap and time efficient [48, 49]. The FLANN model holds the advantage of a single layer perceptron (SLP) network and an MLP network by evading their shortcomings. Here, the adaptive algorithm is the PSO and hence named the model as FLANN-PSO model. A set of input signals is given to the FLANN-PSO model, and the input of the FLANN model is functionally expanded nonlinearly by using the Chebyshev functional expansion technique. All the weights have been updated by using the PSO algorithm. Simultaneously,

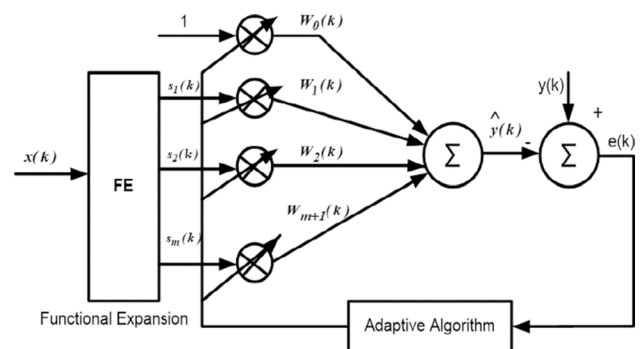


Fig. 4 Structure of the FLANN Model

a set of weights are initialized in between 0 and 1, and it is multiplied with the input signals. The output of each set of weights is compared with the same corresponding desired (output) signal and each set of weights is considered to be one particle. Hence, each set will produce one error signal.

The set of weights whose error is the minimum is considered to be the global particle. The other particles, i.e., other sets of weights, are local particles, and they update their velocity and position according to Eqs. (3) and (4):

$$V_i(d) = wV_i(d) + c_1 * rand * (P_i(d) - X_i(d)) + c_2 * rand * (P_g(d) - X_i(d)) \tag{3}$$

$$X_i(d + 1) = X_i(d) + V_i(d + 1) \tag{4}$$

where $V_i(d)$ and $X_i(d)$ represents the velocity and position of the i th particle, respectively, and $rand$ represents the random number, which is in between $[0,1]$. $P_g(d)$ and $P_i(d)$ are the position of g-best and p-best, respectively. w is the weighting factor, c_1 and c_2 are the constant whose values determine the effect of social and cognitive components.

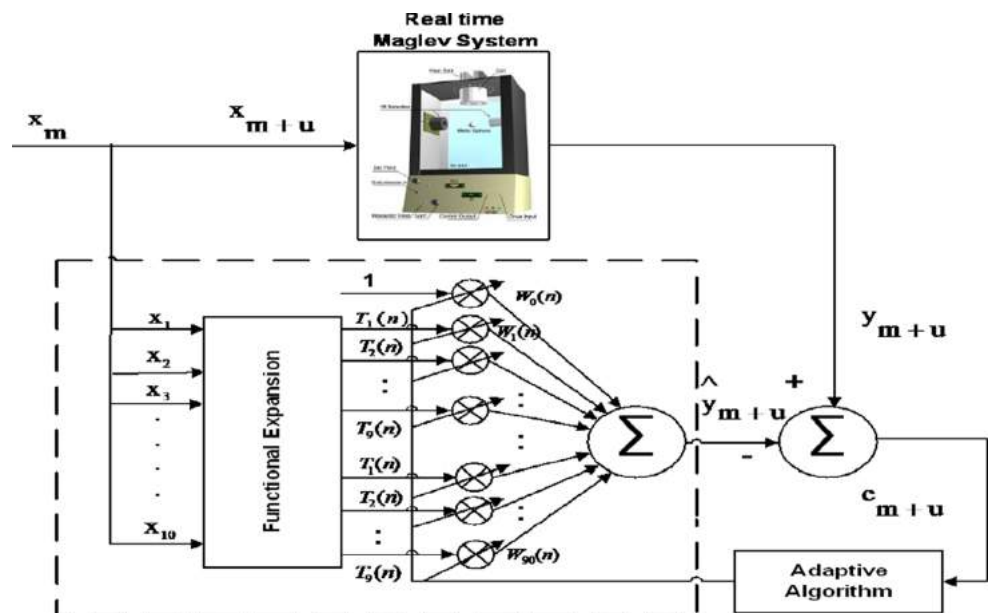
Once all the particles/set of weights are updated, again the error (i.e., objective function) will be calculated by using the new sets of weights. According to the minimum error, the respective weights will be saved so that it can be compared with the previous minimum error. If the current error is lesser than the previous one, then the current value is saved or the previous one. This process is repeated iteratively for a predefined number of times. After a certain epoch, the change of error is saturated, and then, the program is terminated. Finally, the optimum weight is reported. The FLANN network having this optimum weight is called as the trained network, and suitable for testing in the test data.

5 Proposed TLBO-Based FLANN Model

This article presents the metaheuristic TLBO technique based on the teaching and learning methodology, which helps to update the weights of FLANN [29]. The TLBO algorithm simulates a classroom like environment where the number of students is the population whose level of knowledge is considered as the possible solution set of the problem. Hence, the knowledge is defined by its objective function in the problem. The students in a classroom learn mainly through two processes; one through the teacher, and other by interacting between themselves. Thus, TLBO has two phases (a) the teacher phase and (b) the learner phase. In the ‘teacher phase,’ the learner group learns from the teacher, and in the ‘learner phase’ they learn by having discussions with one another. The most knowledgeable person in the classroom is considered as the teacher who shares his or her knowledge with the learners, and at every iteration, the best learner is considered as a teacher. Different designed variables of the optimization problem are analogous to the different subjects offered to the students (learners). The results (grade) of each learner are equivalent to the fitness of the problem. The teacher tries to enhance the knowledge of all the learners in accordance with his or her capability. The transfer of knowledge also depends on the capability of the students (learners).

A set of input signals having window size ‘ u ,’ i.e., $\{x_1, x_2, x_3, \dots, x_u\}$, is given to the proposed FLANN-TLBO model and again, and the input of the FLANN model is functionally expanded nonlinearly by using the Chebyshev functional expansion technique. Simultaneously, random sets of weights (equals to number of expanded inputs of

Fig. 5 Proposed FLANN model for identification of Maglev system



the FLANN) are initialized between 0 and 1. Each set is multiplied by the expanded input signals. Then, the output of the FLANN is compared with the desired signal. Hence, it results a set of error signal $\{e_1, e_2, e_3, \dots, e_u\}$.

Maglev plant input can be expanded by using Chebyshev expansion by the following mathematical form [46],

$$T_i(n) = \begin{bmatrix} T_1 \\ T_2 \\ T_3 \\ \vdots \\ T_k \end{bmatrix} = \begin{bmatrix} 1 \\ x_1 \\ 2x_1^2 - 1 \\ \vdots \\ 2x_k T_k(x_k) - T_{k-1} x_k \end{bmatrix} \text{ For } k > 2 \quad (5)$$

$$\hat{y}_{m+u} = \sum_{i=1}^{Q-1} T_i(n) w_i(n) \quad (6)$$

Here, x_m is the input, \hat{y}_{m+u} is the output of the FLANN model, $T_i(n)$ is the expanded input using Chebyshev expansion and $w_i(n)$ is the weight vector having Q no. of elements. Equation (6) shows the output of the proposed model as shown in Fig. 5. The weights set connected with the FLANN model is optimized by the TLBO algorithm to achieve desired response and the error is

$$e_{m+u} = y_{m+u} - \hat{y}_{m+u} \quad (7)$$

Fig. 6 Flowchart of FLANN-TLBO network

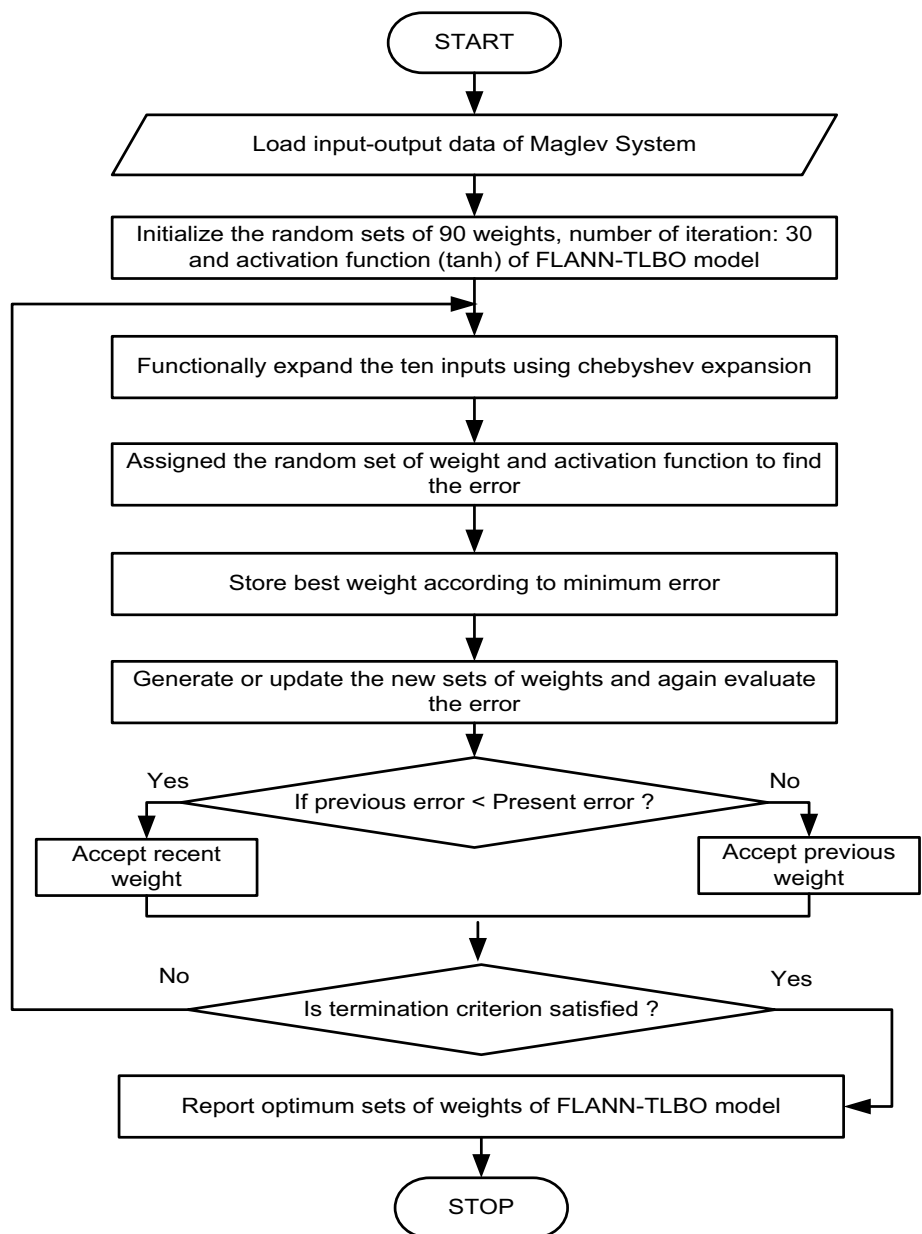


Fig. 7 Model of Fuzzy PID controller

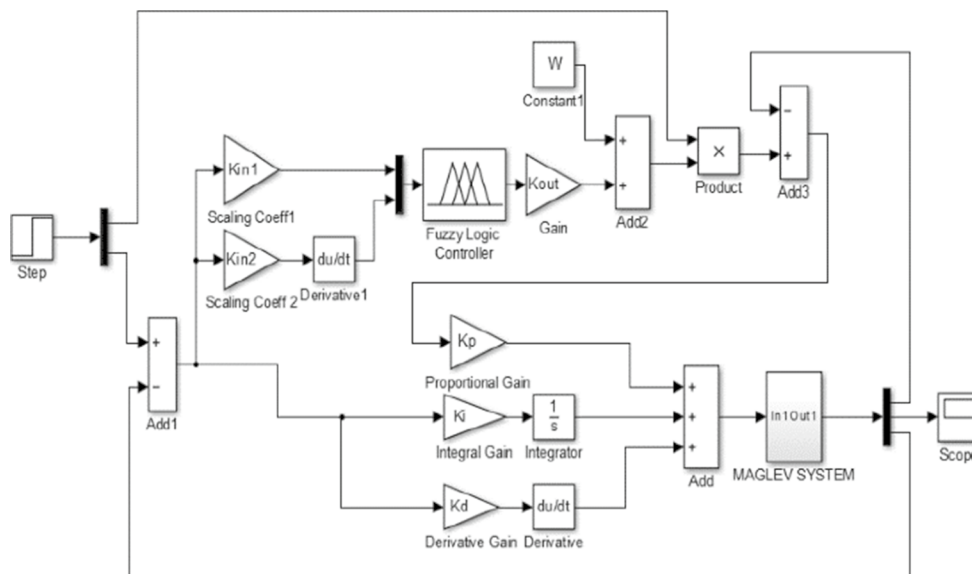


Table 2 Basic rule table for FIS

Row represents the error (e) and column represents the derivative of e					
	NB	NS	Z	PS	PB
NB	NVB	NB	NM	NS	Z
NS	NB	NM	NS	Z	PS
Z	NM	NS	Z	PS	PM
PS	NS	Z	PS	PM	PB
PB	Z	PS	PM	PB	PVB

Table 3 Linguistic variables of FIS

POSITIVE VERY BIG	PVB
POSITIVE BIG	PB
POSITIVE MEDIUM	PM
POSITIVE SMALL	PS
ZERO	Z
NEGATIVE SMALL	NS
NEGATIVE MEDIUM	NM
NEGATIVE BIG	NB
NEGATIVE VERY BIG	NVB

Here, the total input (m) = 1, 2, ..., $n-u$, input window size of FLANN (u) = 10 and n is the no. of expanded input.

5.1 Teaching phase

A teacher tries to enhance the performance of all the students in the class. Considering a class of n students (population size), m subjects (number of design variables) the mean

Fig. 8 Membership function of input variable

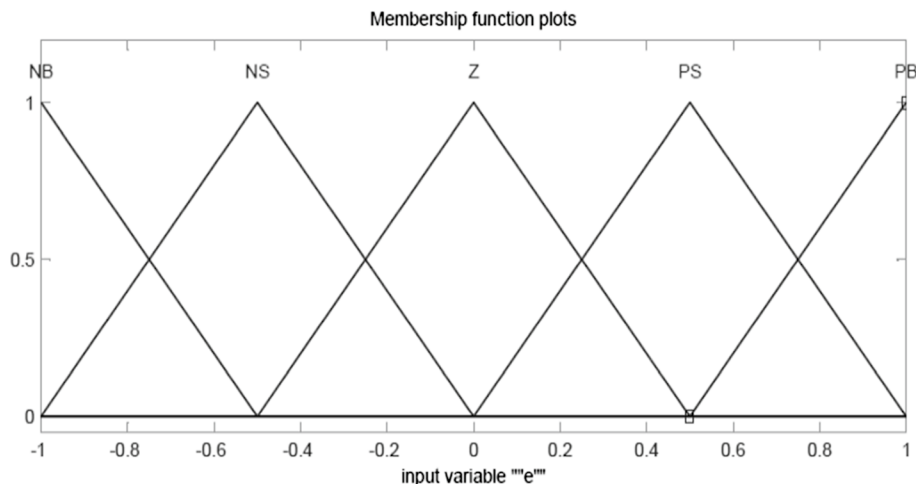
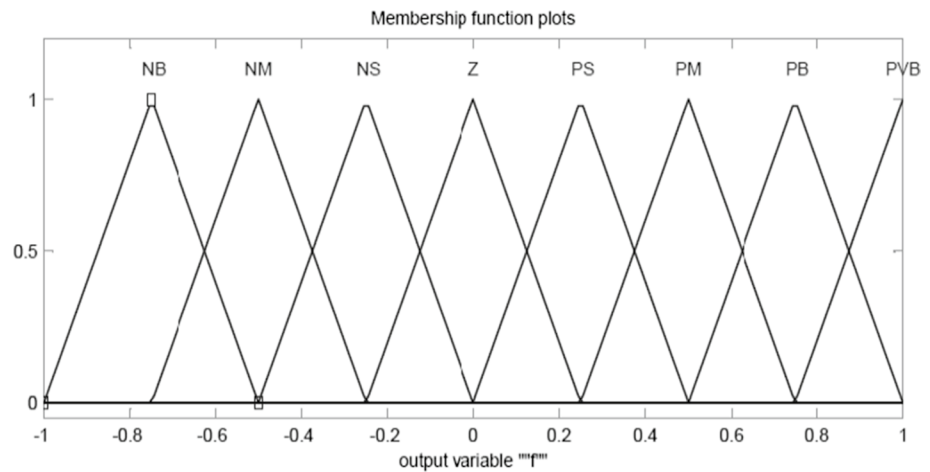


Fig. 9 Membership function for output variable



result of students in subject j can be denoted as M_{ji} for the i th iteration. Let the overall best result of all the subjects of the whole population obtained for the learner number be denoted by $kbest$. The teacher is the most knowledgeable person having the highest fitness value in the class. So, the teacher tries to improve the results of the students, for this a correction variable derived from the difference between the

mean results of the k th student and the teacher in subject j is defined as

$$DM_{j,k,i} = rd \times (W_{j,kbest,i} - T_f \times M_{ji}) \tag{8}$$

where DM correspond to the difference of mean, $W_{j,kbest,i}$ is the result of the teacher in j th subject, rd is a random number between 0 and 1. T_f is called as the teaching factor whose value is either 1 or 2. The teaching factor is defined randomly as

$$T_f = round[1 + rd(0, 1) * (2 - 1)] \tag{9}$$

The solutions are updated as

$$W_{j,k,i}^{new} = W_{j,k,i} + DM_{j,k,i} \tag{10}$$

where $W_{j,k,i}^{new}$ is the updated result value of the k th student in the j th subject at the i th iteration and $W_{j,k,i}$ is the existing result. However, the updated result will be accepted if it satisfies the boundary condition, else it has to be replaced by the limiting boundary value. Also, it should have a better fitness than that of the existing values; otherwise, it need not be replaced. This updated value will act as input to the learner phase (Fig. 5).

5.2 Learner phase

The individual learner enhanced his or her own knowledge by interacting with his/her classmates apart from learning from the teacher. It is a convention that a learner will learn from another learner if the other learner's knowledge is more than his or her. In this phase, two copies p and q are selected randomly such that $x'_{totalpi} \neq x'_{totalqi}$, i.e., the total results as updated in the teacher phase do not match. Then, their results are updated as:

$$x''_{jpi} = x'_{jpi} + r_i(x'_{jpi} - x'_{jq_i}); \text{ if } x'_{totalpi} < x'_{totalqi} \tag{11}$$

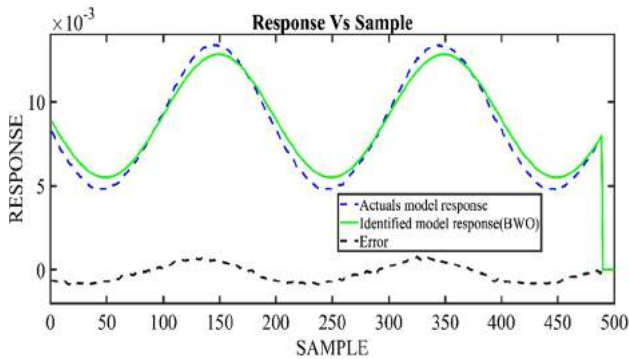


Fig. 10 Identified model response with FLANN-BWO

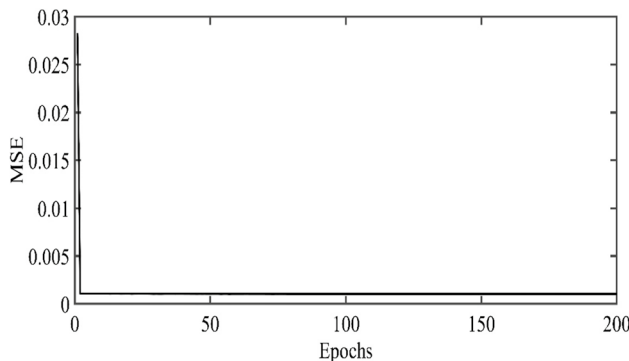


Fig. 11 MSE plot of FLANN-BWO

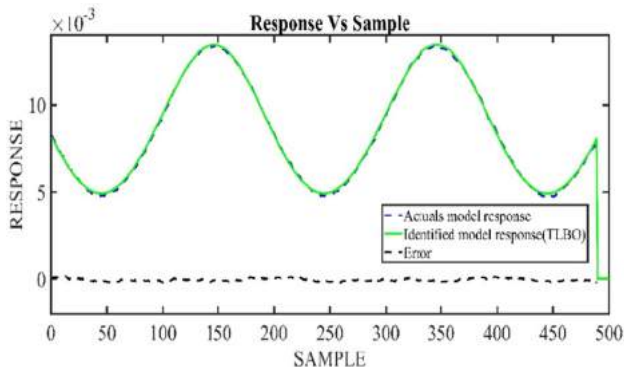


Fig. 12 Identified model response with FLANN-TLBO

$$x''_{jpi} = x'_{jpi} + r_i(x'_{jq_i} - x'_{jpi}); \text{ if } x'_{totalpi} > x'_{totalqi} \quad (12)$$

where x''_{jpi} is accepted if its fitness value is better than that of x'_{jpi} ; further, x''_{jpi} should satisfy the boundary condition. If the boundary condition is not satisfied, it should be replaced by x'_{jpi} .

The 1st set of updated weights and error values is stored for the forthcoming assessment. Again, the TLBO is applied to update the next set of weights and matched with the previous value of weights. The best set of weights, i.e., the set of weight having minimum error is considered to be the teacher, and the other sets are learners. The parameter of the proposed model undergoes the teaching and learning phase of TLBO to update the weights of the FLANN network. This process has been repeated until the error is less than the threshold value. The flowchart describes the detailed process of the TLBO-based FLANN model as shown in Fig. 6.

6 Design of the Fuzzy PID Controller

The universally accepted PID controller is an important tool for industrial control and automation, due to its reliability and adaptability [50]. It has the capability to handle

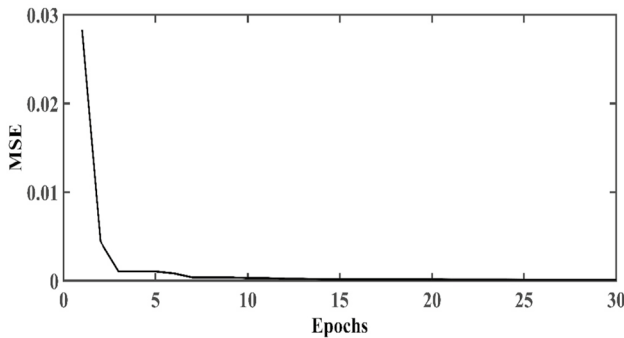


Fig. 13 MSE plot of FLANN-TLBO

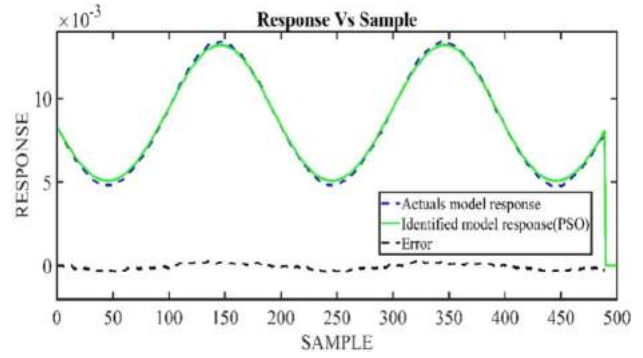


Fig. 14 Identified model response with FLANN-PSO

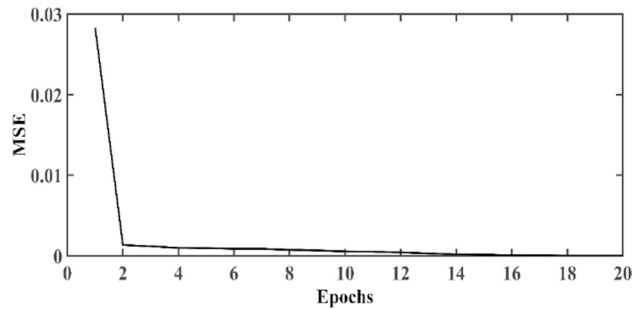


Fig. 15 MSE plot of FLANN-PSO

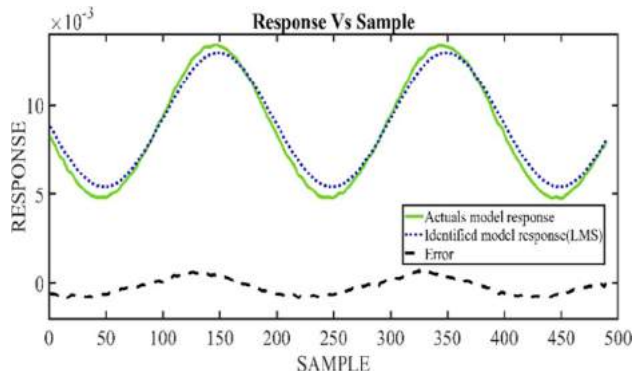


Fig. 16 Identified model response with FLANN-LMS

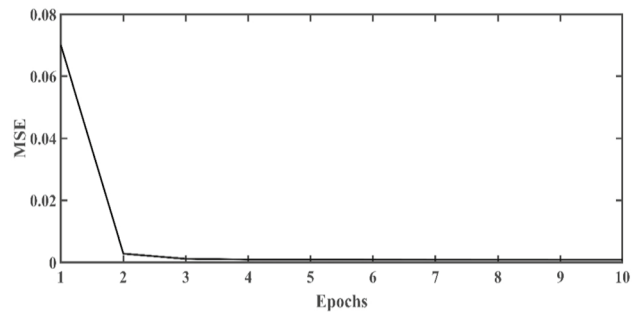


Fig. 17 MSE plot of FLANN-LMS

Table 4 Comparative results of identified model of Maglev system

Model	No. of iteration	Average MSE	CPU time (in s)	Big O Notation
FLANN-BWO	100	2.28E-07	382.422	$O(Ni * (Ps * Fe) * Nr)$
FLANN-TLBO	30	2.7498E-08	462.02	$O(Ni * (Ps * Fe) * Nr)$
FLANN-PSO	20	1.3945E-08	782.43	$O(Ni * (Ps * Fe) * Nr)$
FLANN-LMS	10	2.47E-07	4.15	$O(Ni * Nr)$
MLP-BP	20	1.1470E-07	8.96	$O(Ni * Nr)$

all the shortcomings of any other controller, and is found to be suitable for many of the industrial requirements. However, due to the high nonlinearity and uncertainty present in different systems, the performance of the PID gets degraded. To avoid these bottlenecks and enhance the capability of the PID controller, a fuzzy technique has been incorporated with the PID controller by researchers [51].

The control law associated with PID is as follows:

$$u(t) = k_p e(t) + k_i \int e(t)dt + k_d \frac{de(t)}{dt} \tag{13}$$

where k_p is the proportional gain, k_i is the integral gain, k_d is the derivative gains, $e(t)$ is the error signal and $u(t)$ is the control input.

The intervening system is fuzzified with two inputs, i.e., the system error (e) and derivative of error (\dot{e}), obtained from coefficients, K_{in1} and K_{in2} as shown in Fig. 7. These two values match the values between -1 and 1 . This leads to assign the membership function in a definite manner using the rule Table 2 and the linguistic variable Table 3.

For each input, five membership functions are chosen and assigned. However, for the output, nine triangular membership functions have been defined from -1 to 1 as shown in Figs. 8 and 9, and found from the coefficient K_{out} .

This Fuzzy PID controller has been utilized for validating our identified model. This controller has been implemented to the identified model, and in the real-time Maglev plant. The responses of the identified model and the actual model are compared to investigate the performance of the proposed.

7 Simulation Study

The algorithms were executed in the *Acer Aspire V* system, Window 10 OS, Intel® Core™ i5-3337U CPU @ 1.80 GHz processor, RAM of 8 GB and in a MATLAB environment. Five different neural network models, i.e., the MLP trained by BP, and FLANN networks trained by the LMS, PSO,

TLBO and BWO algorithms have been implemented for the comparative analysis.

7.1 Performance Analysis

All the possible functional expansions are implemented, and we found that the Chebyshev functional expansion model is found to be the most effective in our application. Hence, in our study, we have utilized the Chebyshev expansion in all the four FLANN models for reasonable comparison. The error signal which is the difference between the desired signal and the output of the FLANN network is considered to be the cost function. The following parameter have been considered for the identification of Maglev system using different algorithms.

Table 5 Comparative results of MSE of various optimization techniques for 20 independent test runs

Sl. no.	MSE				
	BWO	TLBO	PSO	LMS	MLP-BP
1	1.99E-07	6.14E-08	1.20E-07	2.71E-07	1.81E-07
2	2.59E-07	1.16E-08	5.83E-08	2.29E-07	1.12E-07
3	2.72E-07	7.93E-09	1.34E-07	2.23E-07	1.50E-08
4	2.25E-07	8.69E-09	4.04E-08	2.85E-07	1.43E-07
5	2.90E-07	2.01E-08	1.46E-07	2.64E-07	2.16E-07
6	2.28E-07	2.82E-08	1.56E-07	2.36E-07	2.56E-07
7	2.67E-07	2.47E-08	1.04E-07	3.02E-07	2.12E-07
8	2.02E-07	5.15E-08	1.33E-07	3.39E-07	3.30E-08
9	2.88E-07	5.66E-08	1.26E-07	1.72E-07	1.95E-07
10	2.44E-07	3.82E-08	1.56E-07	2.73E-07	1.99E-07
11	2.59E-07	4.77E-09	1.36E-07	2.24E-07	1.68E-07
12	2.62E-07	2.29E-07	1.19E-07	1.73E-07	2.27E-07
13	2.94E-07	7.82E-08	1.38E-07	2.45E-07	1.57E-07
14	2.38E-07	1.42E-07	6.61E-08	2.43E-07	1.89E-07
15	2.71E-07	3.40E-08	3.87E-08	3.09E-07	2.76E-07
16	2.75E-07	6.10E-09	8.38E-08	2.80E-07	3.30E-07
17	3.10E-07	5.90E-09	1.51E-08	1.72E-07	1.57E-07
18	2.83E-07	6.69E-08	1.02E-07	1.78E-07	3.80E-08
19	2.17E-07	6.36E-08	1.18E-07	2.28E-07	1.94E-07
20	1.71E-07	4.84E-09	1.26E-07	3.02E-07	2.86E-07

Table 6 The best sets of weight from FLANN-TLBO model ($W_1 - W_{90}$)

W_1	0.435779	W_{19}	0.827147	W_{37}	0.155891	W_{55}	0.264486	W_{73}	0.914621
W_2	0.682444	W_{20}	0.698345	W_{38}	0.112564	W_{56}	0.82409	W_{74}	0.79115
W_3	0.773512	W_{21}	0.320443	W_{39}	0.492218	W_{57}	0.25293	W_{75}	0.958493
W_4	-0.00529	W_{22}	0.953393	W_{40}	0.953885	W_{58}	0.938268	W_{76}	0.787987
W_5	0.781096	W_{23}	0.037445	W_{41}	0.334814	W_{59}	0.358586	W_{77}	0.16796
W_6	0.500079	W_{24}	0.306598	W_{42}	0.579984	W_{60}	0.204795	W_{78}	0.98138
W_7	-0.03474	W_{25}	0.249404	W_{43}	0.218814	W_{61}	0.258882	W_{79}	1.066246
W_8	0.146218	W_{26}	0.633354	W_{44}	0.883247	W_{62}	0.623441	W_{80}	0.81099
W_9	0.414602	W_{27}	0.663029	W_{45}	0.387094	W_{63}	0.480284	W_{81}	0.889997
W_{10}	0.875391	W_{28}	0.054693	W_{46}	0.637974	W_{64}	0.219913	W_{82}	0.825227
W_{11}	0.524488	W_{29}	0.357577	W_{47}	0.831112	W_{65}	0.699049	W_{83}	0.832609
W_{12}	0.787741	W_{30}	0.31339	W_{48}	1.022958	W_{66}	0.453451	W_{84}	0.156097
W_{13}	0.303452	W_{31}	0.514109	W_{49}	1.091364	W_{67}	0.417878	W_{85}	0.969134
W_{14}	0.710593	W_{32}	0.577152	W_{50}	0.679307	W_{68}	0.785315	W_{86}	0.955765
W_{15}	0.036209	W_{33}	0.622466	W_{51}	0.270734	W_{69}	0.153927	W_{87}	0.484031
W_{16}	0.281128	W_{34}	0.268449	W_{52}	0.281422	W_{70}	0.625255	W_{88}	0.798993
W_{17}	0.050205	W_{35}	0.672411	W_{53}	0.389655	W_{71}	0.621751	W_{89}	0.140657
W_{18}	0.100993	W_{36}	0.648091	W_{54}	0.851318	W_{72}	0.248435	W_{90}	0.420589

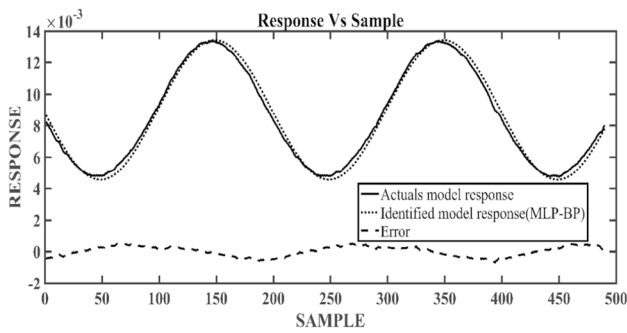


Fig. 18 Identified model response with MLP-BP

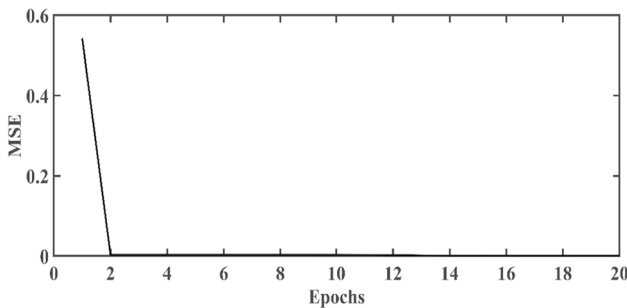


Fig. 19 MSE plot of MLP-BP

In FLANN-LMS: Learning rate (μ): 0.6, No. of iteration (N_i): 10, No. of weights: 90 and Activation function: *tanh*.

In FLANN-PSO: Learning rate (μ): 0.6, No. of iteration (N_i): 20, No. of feature (Fe): 20, cognitive parameter:

$c_1 = c_2 = 2$, Population size (Ps): 45, Inertia rate: 0.9, No. of weights: 90 and Activation function: *tanh*.

In FLANN-TLBO: Population size (Ps): 45, No. of iteration (N_i): 30, No. of feature (Fe): 20, No. of weights: 90 and Activation function: *tanh*.

In FLANN-BWO: Population size (Ps): 45, No. of iteration (N_i): 100, No. of feature (Fe): 20, Procreating rate (PP): 0.6, Cannibalism rate (CR): 0.675, Mutation rate (PM): 0.4, No. of weights: 90 and Activation function: *tanh*.

In MLP-BP: Learning rate (μ): 0.6, No. of iteration (N_i): 20, No. of layer: 3, Node: 5-3-1, No. of weights: 90 and Activation function: *tanh*.

To study the effectiveness of the proposed model, 5000 samples are taken. In Fig. 11, it is shown that the FLANN-BWO model has an average MSE of $2.28E-07$ after 100 iterations and the corresponding CPU time 382.422 s. The FLANN-TLBO model has an average MSE of $2.7498E-08$ after 30 iterations and CPU time 462.02 s as displayed in Fig. 13. But, the FLANN-PSO and MLP-BP models have average MSE of $1.3945E-08$ and $1.1470E-07$, respectively, after 20 iterations each and corresponding CPU time is 782.43 s and 8.96 s, respectively, as presented in Figs. 15 and 19. The gradient-based FLANN-LMS model shown in Fig. 17 has an average MSE of $2.47E-07$ after 10 iterations and CPU time of 4.15 s, which is the lowest among others. By taking the proposed model with different bio-inspired algorithms, the value of MSE has been reduced from $1.1470E-07$ to $2.7498E-08$, as listed in Table 4. After training of the proposed model, the best set of 90 weights, which represents the identified model of the Maglev system, is listed in Table 6. The fitting and MSE curves of all the models are shown in Figs. 10, 11, 12, 13, 14, 15, 16, 17, 18

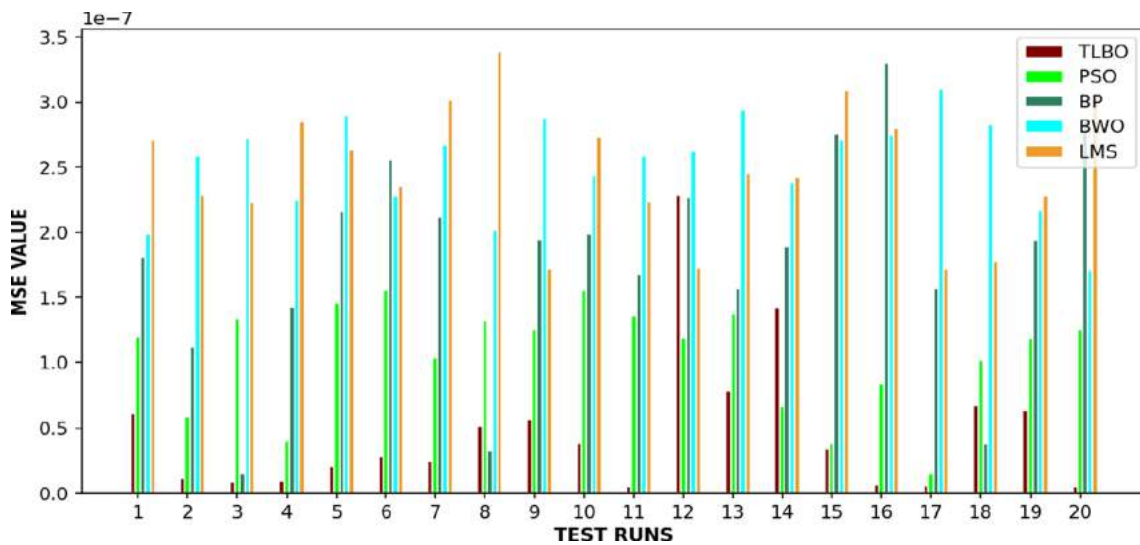


Fig. 20 Comparative plot of MSE in various test runs

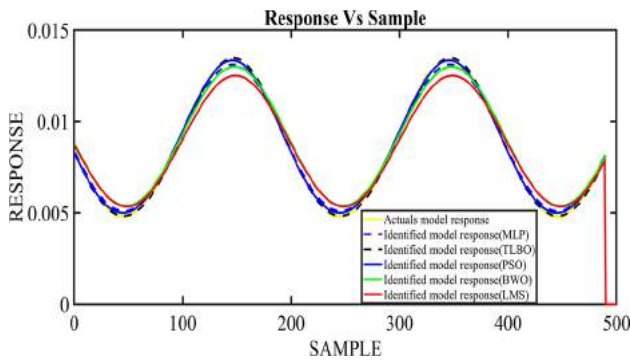


Fig. 21 Comparative identified model response

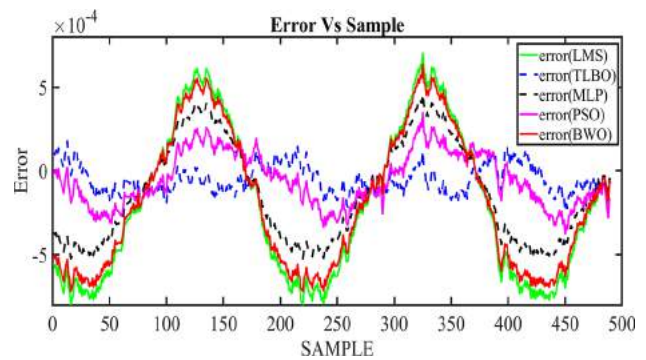


Fig. 22 Comparative error plot

and 19. The comparative results of MSE and the MSE plot are provided for various test runs in Table 5 and Fig. 20.

Here, N_i is the number of iteration, P_s is the number of population, F_e is the number of feature and N_t is the number of input for training. From the Big O Notation, it shows that the FLANN-LMS and MLP-BP have less time complexity than the other three algorithms as shown in Table 4. For investigating the performance objectively, the Mean Squares Error (MSE) is considered as the performance metric. The average values of MSE for all the five models by running them for 20 independent test runs are shown in Table 4. The MSE values for all the models, for each test run in histogram, are shown in Fig. 20. It is clear from Table 4 that the MSE values of the FLANN-TLBO algorithm are lowest as compared to others, which signify the superior performance over the other four competitive networks.

It is depicted from Fig. 21 that the predicted value does not match with the actual output and a very large gap exists.

Hence, the performance is highly unsatisfactory for the FLANN-LMS network. There exists high nonlinearity in the data of the Maglev system, and hence, the result is highly discouraging. The results of the FLANN-TLBO are found to be the most matched one as compared to the other four networks.

From Figs. 21 and 22, it is demonstrated that the response of the FLANN-TLBO model replicates the response of the real-time Maglev system and hence it is the best among all other competitive models. The performance of the algorithm also depends on the number controlling parameters and number of steps associated with weight updation. It is because these two parameters increase the computation time and the computational complexity. From Table 4, it is observed that the FLANN-LMS and FLANN-PSO have taken 4.15 s and 782.43 s CPU time, respectively, which are the lowest and the highest values. The recently developed BWO algorithm-based FLANN network required 382.422 s. The LMS algorithm have one step weight updation with one

Fig. 23 Control of actual Maglev system and identified model using Fuzzy PID controller

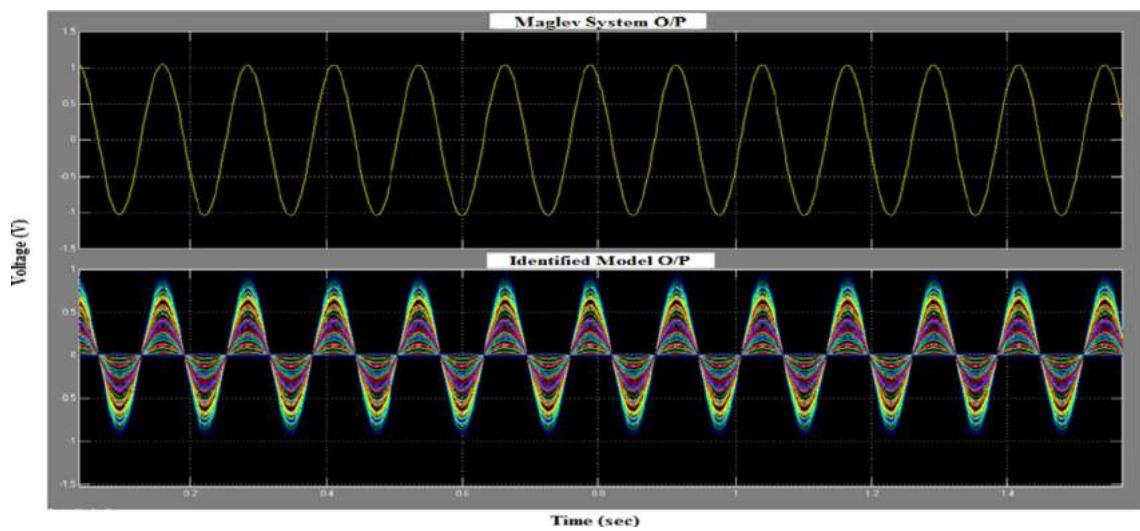
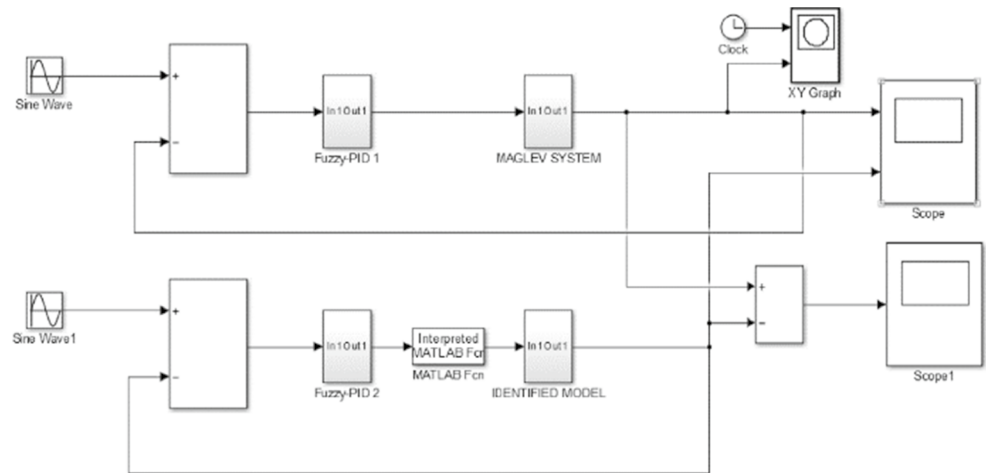


Fig. 24 Comparative results of Maglev system and identified model with a Fuzzy PID controller

Table 7 Minimum wins needed for the two-tailed sign test at $\alpha = 0.05$ and $\alpha = 0.01$

No. of cases	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
$\alpha = 0.05$	5	6	7	7	8	9	9	10	10	11	12	12	13	13	14	15	15	16	17	18	18
$\alpha = 0.01$	5	6	6	7	7	8	9	9	10	10	11	12	12	13	13	14	14	15	16	16	17

Table 8 Critical values obtained for the two-tailed sign tests at $\alpha = 0.05$ and $\alpha = 0.01$ using MSE metric as a triumphant parameter

TLBO	BWO	PSO	LMS	MLP-BP
Wins (+)	20	18	19	17
Loss (-)	0	2	1	3
Detected difference	$\alpha = 0.05$	$\alpha = 0.05$	$\alpha = 0.05$	$\alpha = 0.05$

Table 9 Sign test using MSE metric as a triumphant parameter

Comparison	<i>p</i> value	<i>h</i> value
TLBO with BWO	0.0008	1
TLBO with PSO	0.0004	1
TLBO with LMS	0.0009	1
TLBO with BP	0.0026	1

Table 10 Wilcoxon signed test using MSE metric as a triumphant parameter

Comparison	<i>P</i> value	<i>h</i> value
TLBO with BWO	0.0001	1
TLBO with PSO	0.0025	1
TLBO with LMS	0.0001	1
TLBO with BP	0.0003	1

Table 11 Friedman test rank table

Methods	BWO	TLBO	PSO	LMS	BP
Mean ranks	17	5.2	8.2	17.2	12.4

controlling parameter, the PSO have one step weight updation with three controlling parameters and the MLP have one step weight updation with three controlling parameters. The recently developed BWO algorithm have two steps weight updation with three controlling parameters for which it takes higher time and computational complexity. The FLANN-TLBO model involves two-updation process during the teaching and the learning phases, and hence, it takes more time of 462.02 s.

7.2 Validation of Identified Model

The best identified FLANN-TLBO model is chosen for the validation purpose. The identified Maglev system obtained from the optimal 90 weights is shown in Table 6. This model is controlled and validated by using Fuzzy PID controller with proper choice of the controller parameters. The FLANN-TLBO model having the optimal set of weights is shown in Table 6. The Fuzzy PID controller is used to control both the actual Maglev and the identified model to investigate the response as shown in Fig. 23.

The range of membership functions of the Fuzzy PID controller is defined from -1 to 1 , as shown in Figs. 8 and 9. The best responses are obtained after proper tuning of the Fuzzy PID controller with the values of k_p, k_i and k_d are $-4, -2$ and -0.2 . From Fig. 24, it has been observed that the Fuzzy PID-controlled identified model and the actual Maglev system exhibit the same response.

Table 12 Friedman test parameter

Source	Sum of square (SS)	Degree of freedom (DOF)	Mean square (MS)	Chi-square	Critical value (<i>p</i>)
Columns	141.1	4	35.275	56.44	1.6214E-11
Error	58.9	76	0.775		
Total	200	99			

7.3 Nonparametric Statistical Tests

To validate the dominance of the FLANN-TLBO network, the pairwise sign test and Wilcoxon signed-rank test are carried out. In fact, the sign test and the Wilcoxon signed-rank test are two well-known nonparametric statistical tests proposed for pairwise comparison of the two heuristics approaches. Here, we have carried out the test for 20 runs of each algorithm to justify a fair comparison. The results are listed in Table 8 by considering the average value of MSE as the winning parameter. The minimum number of wins required to obtain $\alpha = 0.05$ and $\alpha = 0.01$ levels of significance for one algorithm over another is shown in Table 7. It is observed in Table 5 that the FLANN-TLBO model shows dominance over all the three other models with a significance of $\alpha = 0.05$.

It is observed from the performance measures of Table 8 that the TLBO shows a significant improvement over the BWO, PSO, LMS and MLP-BP algorithms with a level of significance $\alpha = 0.05$ by taking the detection rate as the winning parameter, and the *p* value and *h* value for the sign test using the MSE metric as a triumphant parameter listed in Table 9. The *p* value and *h* value represent the superiority of the algorithm over the other competitive algorithms. If the *p* value is less than the level of significance $\alpha = 0.05$ and *h* value is 1, then the proposed algorithm is superior over the other and the null hypothesis can be rejected. If the *p* value is greater than the level of significance $\alpha = 0.05$ and *h* value is 0, then the proposed algorithm is considered to be inefficient. Similarly, the Wilcoxon signed-rank test, which is similar to the paired *t* test in statistical procedure, and normally applied to detect the dominance behaviors between the two algorithms, is also performed. The performance comparison of all the algorithms is listed in Table 10. The results presented in Tables 9 and 10 reveal the superiority of the TLBO over other competitive algorithms.

To study the supremacy and repeatability of the obtained response of the network a nonparametric Friedman test is also performed by using the MATLAB. Table 11 shows the average rank of the different networks used for identification, which signifies that the lower rank networks have higher accuracy and performance. The Friedman test parameters are given in Table 12, and the critical value is obtained as $1.6214E-11$ from the Friedman test. A null hypothesis concept emerges, if the critical value is less than the significance

level, i.e., $\alpha = 0.05$ and it can be rejected. Hence, the dominance of the proposed algorithm over other competitive algorithms has been confirmed by performing the sign test, Wilcoxon signed-rank test and Friedman test. A null hypothesis concept comes, if the critical value (p value) is less than the significance level, i.e., $\alpha = 0.05$ and it can be rejected. The result obtained reveals the supremacy of the TLBO over the others.

8 Conclusion

This article proposed a FLANN-TLBO model that yields improved identification and implementation of the Maglev plant. The performance of the TLBO-based FLANN model is compared with that of the other ANN-based models, i.e., MLP-BP, FLANN-LMS, FLANN-PSO and FLANN-BWO. The estimated models have been compared in terms of MSE, CPU time, and the response matching capability of the Maglev system. From the simulations, it is perceived that the proposed FLANN-TLBO model provides superior identification model of the actual Maglev system. The validation of the proposed FLANN-TLBO model is carried out by comparing its performance with the actual Maglev system under identical conditions. The results demonstrate improved response matching of the identified model and the actual system. Moreover, the statistical tests validate the dominance of the FLANN-TLBO network over others. The outcomes of statistical testing reveal the supremacy of the TLBO algorithm in comparison with other competitive algorithms with a significance level of $\alpha = 0.05$. Further, other variants of the neural network and nature-inspired algorithm can be applied for achieving better models of complex systems.


References

- Padoan, A.; Astolfi, A.: Nonlinear system identification for autonomous systems via functional equations methods. In: American Control Conference, pp. 1814–1819 (2016)
- Subudhi, B.; Ieee, S.M.; Jena, D.: Nonlinear system identification of a twin rotor MIMO system. In: IEEE TENCON, pp. 1–6 (2009)
- Weng, B.; Barner, K.E.: Nonlinear system identification in impulsive environments. *IEEE Trans. Signal Process.* **53**, 2588–2594 (2005)
- Forrai, A.: System identification and fault diagnosis of an electromagnetic actuator. *IEEE Trans. Control Syst. Technol.* **25**, 1028–1035 (2017). <https://doi.org/10.1109/TCST.2016.2582147>
- Mondal, A.; Sarkar, P.: A unified approach for identification and control of electro-magnetic levitation system in delta domain. In: International Conference on Control, Instrumentation, Energy and Communication, pp. 314–318 (2016)
- Srivatava, S.; Gupta, M.: A novel technique for identification and control of a non linear system. In: International Conference on Computational Intelligence and Networks, pp. s172–176 (2016). <https://doi.org/10.1109/CINE.2016.37>
- Wen, S.; Wang, Y.; Tang, Y.; Xu, Y.; Li, P.; Zhao, T.: Real-time identification of power fluctuations based on LSTM recurrent neural network: a case study on Singapore Power System. *IEEE Trans. Ind. Inf.* **15**, 5266–5275 (2019). <https://doi.org/10.1109/tii.2019.2910416>
- Alqahtani, A.; Marafi, S.; Musallam, B.; El, N.; Abd, D.; Khalek, E.: Photovoltaic Power Forecasting Model Based on Nonlinear System Identification Modèle de prévision de puissance photovoltaïquebasésur l'. *identification de système non-linéaire* **39**, 243–250 (2016)
- Nanda, S.J.; Panda, G.; Majhi, B.: Improved identification of nonlinear dynamic systems using artificial immune system. In: IEEE Conference and Exhibition on Control, Communications and Automation, pp. 268–273 (2008). <https://doi.org/10.1109/INDCON.2008.4768838>
- Patra, J.; Pal, R.; Chatterji, B.N.; Panda, G.: Identification of nonlinear dynamic systems using functional link artificial neural networks. *IEEE Trans. Syst.* **29**, 254–262 (1999)
- Majhi, B.; Panda, G.: Robust identification of nonlinear complex systems using low complexity ANN and particle swarm optimization technique. *Expert Syst. Appl.* **38**, 321–333 (2011). <https://doi.org/10.1016/j.eswa.2010.06.070>
- Han, M.: Robust Structure Selection of Radial Basis Function Networks for Nonlinear System Identification (2014)
- Chen, W.: Nonlinear System Identification Based on Radial Basis Function Neural Network Using Improved Particle Swarm Optimization, pp. 409–413 (2009). <https://doi.org/10.1109/ICNC.2009.233>
- Kumpati, S.N.; Kannan, P.: Identification and control of dynamical systems using neural networks. *IEEE Trans. Neural Netw.* **1**, 4–26 (1990)
- Pao, Y.H.: Adaptive Pattern Recognition and Neural Networks. Addison-Wesley, Reading (1989)
- Mallikarjuna, B.; Viswanathan, R.; Naib, B.B.: Feedback-based gait identification using deep neural network classification. *J. Crit. Rev.* **7**, 661–667 (2020). <https://doi.org/10.31838/jcr.07.04.125>
- Vora, D.R.; Rajamani, K.: A hybrid classification model for prediction of academic performance of students: a big data application. *Evol. Intell.* (2019). <https://doi.org/10.1007/s12065-019-00303-9>
- Guo, Y.; Wang, F.; Lo, J.T.H.: Nonlinear system identification based on recurrent neural networks with shared and specialized memories. In: Asian Control Conference 2018-January, pp. 2054–2059 (2018). <https://doi.org/10.1109/ASCC.2017.8287491>
- Wang, Z.; Gu, H.: Nonlinear system identification based on genetic algorithm and grey function. In: IEEE International Conference on Automation and Logistics, pp. 1741–1744 (2007)
- Guoqiang, Y.; Weiguang, L.; Hao, W.: Study of RBF neural network based on PSO algorithm in nonlinear system (2015). <https://doi.org/10.1109/ICICTA.2015.217>
- Kang, D.; Lee, B.; Won, S.: Nonlinear system identification using ARX and SVM with advanced PSO. In: IEEE Industrial Electronics Society, pp. 598–603 (2007)
- Panda, G.; Mohanty, D.; Majhi, B.; Sahoo, G.: Identification of nonlinear systems using particle swarm optimization technique. In: IEEE Congress on Evolutionary Computation, pp. 3253–3257 (2007). <https://doi.org/10.1109/CEC.2007.4424889>
- Hayyolalam, V.; PourhajiKazem, A.A.: Black widow optimization algorithm: a novel meta-heuristic approach for solving engineering optimization problems. *Eng. Appl. Artif. Intell.* **87**, 103249 (2020). <https://doi.org/10.1016/j.engappai.2019.103249>
- Kumar, M.; Kulkarni, A.J.; Satapathy, S.C.: Socio evolution & learning optimization algorithm: a socio-inspired optimization



- methodology. *Future Gener. Comput. Syst.* **81**, 252–272 (2018). <https://doi.org/10.1016/j.future.2017.10.052>
25. Gholizadeh, S.; Milany, A.: An improved fireworks algorithm for discrete sizing optimization of steel skeletal structures. *Eng. Optim.* **50**, 1829–1849 (2018). <https://doi.org/10.1080/0305215X.2017.1417402>
 26. Gholizadeh, S.; Ebadijalal, M.: Performance based discrete topology optimization of steel braced frames by a new metaheuristic. *Adv. Eng. Softw.* **123**, 77–92 (2018). <https://doi.org/10.1016/j.advengsoft.2018.06.002>
 27. Gholizadeh, S.; Danesh, M.; Gheytratmand, C.: A new Newton metaheuristic algorithm for discrete performance-based design optimization of steel moment frames. *Comput. Struct.* **234**, 106250 (2020). <https://doi.org/10.1016/j.compstruc.2020.106250>
 28. Rao, R.V.; Savsani, V.J.; Vakharia, D.P.: Teaching-learning-based optimization: a novel method for constrained mechanical design optimization problems. *Comput. Aided Des.* **43**, 303–315 (2011). <https://doi.org/10.1016/j.cad.2010.12.015>
 29. Naik, B.; Nayak, J.; Behera, H.S.: A TLBO based gradient descent learning-functional link higher order ANN: An efficient model for learning from non-linear data. *J. King Saud Univ. Comput. Inf. Sci.* **30**, 120–139 (2018). <https://doi.org/10.1016/j.jksuci.2016.01.001>
 30. Naumovi, M.B.; Veseli, B.R.: Magnetic levitation system in control engineering education. *Autom. Control Robot.* **7**, 151–160 (2008)
 31. Morales, R.; Feliu, V.; Member, S.; Sira-ramírez, H.; Member, S.: Nonlinear control for magnetic levitation systems based on fast online algebraic identification of the input gain. *IEEE Trans. Control Syst. Technol.* **19**, 757–771 (2011)
 32. Balko, P.; Rosinova, D.: Modeling of magnetic levitation system. In: *International Conference on Process Control*. pp. 252–257 (2017)
 33. Liceaga-castro, J.; Hernandez-alcantara, D.; Amezcua-brooks, L.: Nonlinear control of a magnetic levitation system. In: *Electronics, Robotics and Automotive Mechanics Conference*, pp. 391–396 (2009). <https://doi.org/10.1109/CERMA.2009.10>
 34. Magnetic Levitation: Control Experiments Feedback Instruments Limited (2011)
 35. Ghosh, A.; Krishnan, T.R.; Tejaswy, P.; Mandal, A.; Pradhan, J.K.; Ranasingh, S.: Design and implementation of a 2-DOF PID compensation for magnetic levitation systems. *ISA Trans.* **53**, 1216–1222 (2014)
 36. Swain, S.K.; Sain, D.; Kumar, S.; Ghosh, S.: Real time implementation of fractional order PID controllers for a magnetic levitation plant. *Int. J. Electron. Commun.* **78**, 141–156 (2017)
 37. Yaghini, M.; Khoshraftar, M.M.; Fallahi, M.: A hybrid algorithm for artificial neural network training. *Eng. Appl. Artif. Intell.* **26**(1), 293–301 (2013)
 38. Patra, J.C.; Kot, A.C.: Nonlinear dynamic system identification using Chebyshev functional link artificial neural networks. *IEEE Trans. Syst. Man. Cybern.* **32**, 505–511 (2002). <https://doi.org/10.1109/TSMCB.2002.1018769>
 39. Subudhi, B.; Jena, D.: Nonlinear system identification using memetic differential evolution trained neural networks. *Neurocomputing.* **74**, 1696–1709 (2011). <https://doi.org/10.1016/j.neucom.2011.02.006>
 40. Katari, V.; Malireddi, S.; Satya, S.K.; Panda, G.: Adaptive nonlinear system identification using comprehensive learning PSO. In: *International Symposium on Communications, Control and Signal Processing*, pp. 434–439 (2008). <https://doi.org/10.1109/ISCCSP.2008.4537265>
 41. Juang, J.-G.; Lin, B.-S.: Nonlinear system identification by evolutionary computation and recursive estimation method. In: *American Control Conference*, pp. 5073–5078 (2005). <https://doi.org/10.1109/CINE.2015.22>
 42. Puchta, E.D.P.; Siqueira, H.V.; Kaster, M.D.S.: Optimization tools based on metaheuristics for performance enhancement in a gaussian adaptive PID controller. *IEEE Trans. Cybern.* **50**, 1185–1194 (2020). <https://doi.org/10.1109/TCYB.2019.2895319>
 43. Rao, R.V.; Savsani, V.J.; Balic, J.: Teaching-learning-based optimization algorithm for unconstrained and constrained real-parameter optimization problems. *Eng. Optim.* **44**, 1447–1462 (2012). <https://doi.org/10.1080/0305215X.2011.652103>
 44. Kumar, M.; Mishra, S.K.: Teaching learning based optimization-functional link artificial neural network filter for mixed noise reduction from magnetic resonance image. *Bio Med. Mater. Eng.* **28**, 643–654 (2017)
 45. Singh, S.; Ashok, A.; Kumar, M.; Rawat, T.K.: Adaptive infinite impulse response system identification using teacher learner based optimization algorithm. *Appl. Intell.* **49**, 1785–1802 (2018). <https://doi.org/10.1007/s10489-018-1354-4>
 46. Patra, J.C.; Kot, A.C.: Nonlinear dynamic system identification using chebyshev functional link artificial neural network. In: *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 32, pp. 505–511 (2002)
 47. Li, M., He, Y.: Nonlinear system identification using adaptive Chebyshev neural networks. *IEEE International Conference on Intelligent Computing and Intelligent Systems*, pp. 243–247 (2010)
 48. Nanda, S.J., Panda, G., Majhi, B., Tah, P.: Improved Identification of Nonlinear MIMO Plants using New Hybrid FLANN-AIS Model. In: *International Advanced Computing Conference*, pp. 141–146 (2009). <https://doi.org/10.1109/IADCC.2009.4808996>
 49. Kumar, M.; Mishra, S.K.: Particle swarm optimization-based functional link artificial neural network for medical image denoising. In: *Computational Vision and Robotics*, vol. 105–111 (2015)
 50. Arora, A.; Hote, Y.V.; Rastogi, M.: Design of PID controller for unstable system. *Commun. Comput. Inf. Sci.* **140**, 19–26 (2011). https://doi.org/10.1007/978-3-642-19263-0_3
 51. Rastogi, M.A.; Arora, Y.V.H.: Design of Fuzzy Logic Based PID Controller for an Unstable System, Vol. 157, p. 66–571. Springer, Berlin (2011)



ITC 3/50 Information Technology and Control Vol. 50 / No. 3 / 2021 pp. 411-423 DOI 10.5755/j01.itc.50.3.27349	Development of Proposed Ensemble Model for Spam e-mail Classification	
	Received 2020/07/18	Accepted after revision 2021/06/08
	 http://dx.doi.org/10.5755/j01.itc.50.3.27349	

HOW TO CITE: Shrivastava, A. K., Dewangan, A. K., Ghosh, S. M., Singh, D. (2021). Development of Proposed Ensemble Model for Spam e-mail Classification. *Information Technology and Control*, 50(3), 411-423. <https://doi.org/10.5755/j01.itc.50.3.27349>

Development of Proposed Ensemble Model for Spam e-mail Classification

Akhilesh Kumar Shrivastava

Guru Ghasidas Vishwavidyalaya, Bilaspur (C.G.), India

Amit Kumar Dewangan

Dr. C. V. Raman University, Bilaspur (C.G.), India

S. M. Ghosh

Dr. C. V. Raman University, Bilaspur (C.G.), India

Devendra Singh

Guru Ghasidas Vishwavidyalaya, Bilaspur (C.G.), India

Corresponding author: akhilesh.mca29@gmail.com

Spam e-mail documents classification is a very challenging task for e-mail users, especially non IT users. Billions of people using the internet and face the problem of spam e-mails. The automatic identification and classification of spam e-mails help to reduce the problem of e-mail users in managing a large amount of e-mails. This work aims to do a significant contribution by building a robust model for classification of spam e-mail documents using data mining techniques. In this paper, we use Enron1 data set which consists of spam and ham documents collected from Kaggle repository. We propose an Ensemble Model-1 that is an ensemble of Multilayer Perceptron (MLP), Naïve Bayes and Random Forest (RF) to obtain better accuracy for the classification of spam and ham e-mail documents. Experimental results reveal that the proposed Ensemble Model-1 outperforms other existing classifiers as well as other proposed ensemble models in terms of classification accuracy. The suggested and proposed Ensemble Model-1 produces a high accuracy of 97.25% for classification of spam e-mail documents.

KEYWORDS: Ensemble Model, Classification, Data Mining, Spam e-mail, Machine Learning.



Animal detection based on deep convolutional neural networks with genetic segmentation

Ramakant Chandrakar¹ · Rohit Raja² · Rohit Miri¹

Received: 8 January 2021 / Revised: 13 June 2021 / Accepted: 9 July 2021

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2021

Abstract

This paper presents a system for automatic detection and recognition of the animals using Deep CNN with genetic segmentation. In the present work, the grouping of input animal pictures is done with the help of a Convolutional Neural Network is demonstrated. The proposed work is compared with standard recognition methods such as SU, DS, MDF, LEGS, DRFI, MR, GC. The existing methodologies have more error rates because of high false-positive & negative rate detection, hence there is a need for a highly accurate system for animal detection. According to the proposed work, a genetic algorithm is used for the segmentation process, and for classification 3-layers neural network is used. For training and examining the proposed work, a database is created which consists of 100 distinct subjects with 2 classes and 10 pictures in each class. Experimental results are demonstrated as the segmentation using genetic algorithms and the novelty of the proposed method in terms of precision, recall, f-measurement, and MAE. Hence proposed work improves the overall results i.e. precision (99.02%), recall (98.79%), F-Measurement (98.9%), and MAE (0.78%).

Keywords Animal classification · CNN · Saliency map · Genetic algorithm

✉ Rohit Raja
drrohitraja1982@gmail.com

Ramakant Chandrakar
ramakant.chandrakar42@gmail.com

Rohit Miri
rohitmiri@gmail.com

¹ Department of CSE, Dr. C. V. RAMAN University, Bilaspur, India

² Department of IT, Guru Ghasidas Vishwavidyalaya (Central University), Bilaspur, CG, India

An Effective Contour Detection based Image Retrieval using Multi-Fusion Method and Neural Network

Rohit Raja

Guru Ghasidas Vishwavidyalaya: Guru Ghasidas University

Sandeep Kumar (✉ er.sandeepsahratia@gmail.com)

Sreyas Institute of Engineering and Technology <https://orcid.org/0000-0002-4752-7884>

Shilpa Choudhary

Neil Gogte Institute of Technology

Hemlata Dalmia

Sreyas Institute of Engineering and Technology

Research Article

Keywords: YCbCr, HSV Histogram, Contour, CBIR, Sobel Edge Detection, NN

Posted Date: June 1st, 2021

DOI: <https://doi.org/10.21203/rs.3.rs-458104/v1>

License:   This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

 View PDFAccess through **your institution**[Purchase PDF](#)

Computer Communications

Volume 176, 1 August 2021, Pages 146-154

Internet of Things attack detection using hybrid Deep Learning Model

Amiya Kumar Sahu ^a  , Suraj Sharma ^a , M. Tanveer ^b , Rohit Raja ^c [Show more](#)  Outline |  Share  Cite<https://doi.org/10.1016/j.comcom.2021.05.024>[Get rights and content](#)

Abstract

The Internet of Things (IoT) has become a very popular area of research due to its large-scale implementation and challenges. However, security is the key concern while witnessing the rapid growth in its size and applications. It is a tedious task to individually put security mechanisms in each IoT device and update it as per newer threats. Moreover, machine learning models can best utilize the colossal amount of data generated by IoT devices. Therefore, many Deep Learning (DL) based mechanisms have been proposed to detect attacks in IoT. However, the existing security mechanisms addressed limited attacks, and they used limited and outdated datasets for evaluations. This paper presents a novel security framework and an attack detection mechanism using a Deep Learning model to fill in the gap, which will efficiently detect malicious devices. The proposed mechanism uses a Convolution Neural Network (CNN) to extract the accurate feature representation of data and further classifies those by Long Short-Term Memory (LSTM) Model. The dataset used in the experimental evaluation is from twenty Raspberry

FEEDBACK 

Research Article

A Comparative Analysis of Machine Learning Algorithms for Detection of Organic and Nonorganic Cotton Diseases

Sandeep Kumar ¹, Arpit Jain ², Anand Prakash Shukla ³, Satyendr Singh ⁴,
Rohit Raja ⁵, Shilpa Rani ⁶, G. Harshitha ¹, Mohammed A. AlZain ⁷,
and Mehedi Masud ⁸

¹Sreyas Institute of Engineering & Technology, Hyderabad, India

²Teerthanker Mahaveer University, Moradabad, U.P, India

³KIET Group of Institutions, Gaziabad, India

⁴BML Munjal University, Gurugram, India

⁵Central University, Bilaspur, Chhattisgarh, India

⁶Neil Gogte Institute of Technology, Hyderabad, India

⁷Department of Information Technology, College of Computers and Information Technology, Taif University, P.O. Box 11099, Taif 21944, Saudi Arabia

⁸Department of Computer Science, College of Computers and Information Technology, Taif University, P.O. Box 11099, Taif 21944, Saudi Arabia

Correspondence should be addressed to Mehedi Masud; mmasud@tu.edu.sa

Received 2 May 2021; Revised 31 May 2021; Accepted 7 June 2021; Published 17 June 2021

Academic Editor: Dr. Dilbag Singh

Copyright © 2021 Sandeep Kumar et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Cotton is the natural fiber produced, and the commercial crop grown in monoculture on 2.5% of total agricultural land. Cotton is a drought-resistant crop that provides a reliable income to the farmers that grow under the area with a threat from climatic change. These cotton crops are being affected by bacterial, fungal, viral, and other parasitic diseases that may vary due to the climatic conditions resulting in the crop's low productivity. The most prone to diseases is the leaf that results in the damage of the plant and sometimes the whole crop. Most of the diseases occur only on leaf parts of the cotton plant. The primary purpose of disease detection has always been to identify the diseases affecting the plant in the early stages using traditional techniques for better production. To detect these cotton leaf diseases appropriately, the prior knowledge and utilization of several image processing methods and machine learning techniques are helpful.

1. Introduction

Day by day, all over the world, agriculture land is going to be reduced because the population is increasing rapidly and lack of water resources. Disease in the plant is one of those hazards that have to be examined at this stage. In contrast, the isolation of plants from their natural environment is being happened, and they are grown in unusual conditions. Many valuable crops and plants are very vulnerable to disease. They would have a great struggle to survive in nature without human involvement. Yield loss in harvests is

regularly connected with plant illness or factors, such as climate, water availability, and supplement accessibility [1]. To improve the productivity of the crop, environmental factors or product resources such as temperature and humidity are important [2]. An important role is played by the root exudates of the plant, which helps in improving the nutrients of the soil [3]. Compared to their wild relations, cultivated plants are always more flexible to diseases. This is the large numbers of the same species or different kinds, having a similar gene grown together, sometimes over many kilometers of distance.

Broadcasting the Transaction System by Using Blockchain Technology

K. Rohit Kumar,

Assistant Professor, Department of Computer Science and Engineering

Rohit Raja,

Associate Professor, Department of Information Technology

¹Koneru Lakshmaiah Education Foundation, Greenfields, Vaddeswaram, Guntur-522502

²GGV (Central University), Bilaspur, Chhattisgarh.

Abstract:

The process of the Transaction to implement different strategies for processing an application by using decentralized system using Blockchain technology. In this major part of the blockchain Technology is to achieve the target on achieving the scalability and reliability of the component block. The efficiency and accuracy of Network synchronization in order to solve the problems, we improved the block broadcasting protocol which elaborates the data in the given block. In the proposed node the relay a piece of the block will get benefited with financial benefit. Blockchain Technology can be implemented by applying the data sharing our proposed method to speed up the block broadcasting and there synchronization time.

1.INTRODUCTION

Following the emergence of Bitcoin, Blockchain technologies gained much interest from colleges and industry. As the blockchain does not require total confidence between nodes with many unique characteristics, including immutability, due to its particular data structure within the chain, consent algorithms and rewards schemes. Although the Blockchain technology has numerous advantages it does not face the biggest barrier in the large-scale usage, poor efficiency and high latency. Bitcoin develops one block[1] in about 10 minutes with 7 twice (transaction perseconds). Now the consolidated payment networks are being managed by more than 24,000 TPS[2] or three or four orders greater than Bitcoin, such as master cards and visas. The second Ethereum is also recognized as this.

The generation of Blockchain is just 15 TPS[3], already way below the central payment structure. Blockchain Broadcasting Technologies suggested consensus algorithms could tolerated network error[4]. This implies higher competition costs and poorer results than conventional records. Any careful studies were performed to change the blockchain efficiency by enhancing the consensus algorithm. The findings can be classified into four groups. (2) To maintain a fork avoidance[5], the fundamental data layout has been changed[6].2) Decentralization needs have



ScienceDirect

 View PDFAccess through **your institution**[Purchase PDF](#)

Measurement

Volume 172, February 2021, 108882

Detection of lung nodule and cancer using novel Mask-3 FCM and TWEDLNN algorithms

Laxmikant Tiwari ^a, Rohit Raja ^b  , Vineet Awasthi ^c, Rohit Miri ^c, G.R. Sinha ^d, Monagi H. Alkinani ^e , Kemal Polat ^f 

[Show more](#) [Outline](#) | [Share](#)  [Cite](#) <https://doi.org/10.1016/j.measurement.2020.108882>[Get rights and content](#)

Highlights

- To detect lung cancer by using the deep learning algorithm.
- To enhance the lung image using the MC-CLAHE algorithm.
- To detect the lung nodule from the lung images using a new Mask-3FCM algorithm.

Abstract

FEEDBACK 

Multi-Label Classification for Images with Labels for Image Annotation

Swati Jain¹, Md Rashid Mahmood², Rohit Raja³, K.Ramya Laxmi⁴, Akanksha Gupta⁵

¹ Govt. J.Y. Chhattisgarh College, Raipur, India; e-mail : sjcscghed@gmail.com

² ECE department, Guru Nanak Institutions Technical Campus, Hyderabad, India, e-mail : er.mrashid@gmail.com

³ IT Department, GGV Central University, Bilaspur Central University India, e-mail : drrohitraja1982@gmail.com

⁴ SIET Hyderabad, India; e-mail : kunt.ramya@gmail.com

⁵ IT Department, GGV Central University, Bilaspur Central University India; e-mail : akanksha.me2011@gmail.com

ABSTRACT

Images and videos are increasing due to advancement in digital technologies. Annotate the given image for efficient image retrieval and processing it is required identifying a set of objects present in each image. Manual text-based image annotation is very time consuming and expensive and it becomes infeasible with such exponential increase in visual data. Multi-label classification problem generalizes the standard multiclass classification by allowing each instance to be simultaneously assigned into multiple label categories. A key challenge for multi-label classification is label sparsity that is many labels lacks sufficient training instances for building efficient classifiers. Hence, exploiting label dependency can significantly boost classification performance. Most of multi-label method uses binary decomposition of multi-label datasets but uses the same features for training the classifiers which may contain redundant features.

Keywords: Visual Data, Classification, Neural Network.

SAMRIDDHI : A Journal of Physical Sciences, Engineering and Technology, (2020); DOI : 10.18090/samriddhi.v12iS3.25

INTRODUCTION

The collection of digital images is useful if a user can find the images of desired content from it. The content management of pictorial data is an organized way to store and search the images from the database. The objective of the automated system is by using the features of the image contents understand the image [1]. In some cases, the dataset contains the extracted features from images and so the methods perform relatively better compared to datasets from which extraction has to be done. Feature extraction is important as it dominates the rest of the functionality of automatic image annotation [2].

LITERATURE REVIEW

Automatic Image Annotation techniques have been discovered to label the entities present in the image.

Corresponding Author :

Swati Jain, Govt. J.Y. Chhattisgarh College, Raipur, India; e-mail : sjcscghed@gmail.com

How to cite this article :

Jain, S., Mahmood, Md.R., Raja, R., Laxmi, K.R., Gupta, A. (2020). Multi-Label Classification for Images with Labels for Image Annotation. *SAMRIDDHI : A Journal of Physical Sciences, Engineering and Technology, Volume 12, Special Issue (3), 122-127.*

Source of support : Nil

Conflict of interest : None

There have been various techniques used for AIA which can be best implemented using multi-label classification. Hence in order to perform AIA we have to perform multi-label classification of images [3].

Implementation of Multi Sensor and Multi-Functional Mobile Robot for Image Mosaicking

Md Rashid Mahmood¹, Rohit Raja², Akanksha Gupta³, Swati Jain⁴

¹ ECE department, Guru Nanak Institutions Technical Campus, Hyderabad, India; e-mail : er.mrashid@gmail.com

² IT Department, GGV Central University, Bilaspur Central University India, e-mail: drrohitraja1982@gmail.com

³ IT Department, GGV Central University, Bilaspur Central University India; e-mail : akanksha.me2011@gmail.com

⁴ Govt. J. Y. Chhattisgarh College, Raipur, India; e-mail : sjcscghed@gmail.com

ABSTRACT

In this paper a multi-sensory and multi-functional robot will be developed with the features of obstacle avoider and follower, black and white line follower, motion of the robot will be based on clap sound and flash of light followed and avoider robot. The robot will have 3 DOF (Degree of freedom) arm with a camera on the top of that arm through which successive images or videos are taken in any places. In the future, we can use the DTMF receiver by using the mobile phone. This robot can be used for military purpose in surveillances in any environment and weathers which save human's lives and searching purpose also where the human cannot reach. Robot with wireless camera provides numerous applications in spy. This project helps us to follow or detect a person in the building or any critical place without the presence of us. We can see the video of the present state of the building or place on the T.V. Thus, this project provides security.

Keywords: 2DOF, edge avoider, image mosaicking, IR images in continuous manner than images are joints by the sensor and line follower.

SAMRIDDHI : A Journal of Physical Sciences, Engineering and Technology, (2020); DOI : 10.18090/samriddhi.v12iS3.26

INTRODUCTION

Making of multi sensor and multi-functional robot is the main concern of the researchers all over the world, including different software like image masking, surveillances etc. to makes robot perfect and reliable and they are developing various methodologies to overcome the problem. Robotics is the branch of technology that deals with the designs, constructions, operations and of robot, as well as computer systems for their controls, sensory feedbacks, and information processing. "A robot is a reprogrammable, multi-functional designed to move material, parts, tools, or specialized devices, arrangement and operates through variable programmed motions for the presentation of a selection of tasks." Robotics is the technology and discipline of robots, their manufacture, application and design. Robotics must have a working knowledge of electronics, mechanics, and software. The robot

Corresponding Author : Md Rashid Mahmood, ECE department, Guru Nanak Institutions Technical Campus, Hyderabad, India; e-mail : er.mrashid@gmail.com

How to cite this article : Mahmood, Md. R., Raja, R., Gupta, A., Jain, S. (2020). Implementation of Multi Sensor and Multi-Functional Mobile Robot for Image Mosaicking. *SAMRIDDHI : A Journal of Physical Sciences, Engineering and Technology, Volume 12, Special Issue (3), 128-135.*

Source of support : Nil

Conflict of interest : None

is prepared mechanically which is called a kinematic chain. It functions analogously like creature skeleton. This chain contains actuators (muscles), links (bones) and joints.

Image mosaic is one of the key research areas in an image processing. Image mosaicking is ended in

Parametric Linear Subspace Method for Face Recognition and Pose Estimation

Md Rashid Mahmood^{*1}, Ashwala Mohan², Rohit Raja³, Akanksha Gupta⁴

^{1,2} ECE department, Guru Nanak Institutions Technical Campus, Hyderabad, India, e-mail : er.mrashid@gmail.com, ashwala@yahoo.com

^{3,4} IT Department, GGV Central University, Bilaspur Central University India, e-mail: drrohitraja1982@gmail.com, akanksha.me2011@gmail.com

ABSTRACT

A robust face recognition system is described in this paper which is insensitive to human head pose. The combination of Gabor Wavelet Network (GWN) and Parametric Linear Subspace (PLS) method is used for effective and efficient face recognition process. Multiple feature extraction is done and various steps of pose estimation carried out using new algorithms. In this work, the complete algorithm has been carried out in various stages. Initially, source images are applied and flag is set. In the next step, image is converted to gray scale image and the given image's extension is converted into portable gray map. calculate the analysis mapping that is estimate the pose of head from given image, given image is compared with the database, decision is made according to the matched or unmatched templates, above steps are repeated until matching of last database template.

Keywords: Face recognition, GWN, PLS, Pose estimation, Classification of Face Recognition, LPCMAP.

SAMRIDDHI : A Journal of Physical Sciences, Engineering and Technology, (2020); DOI : 10.18090/samriddhi.v12iS3.29

INTRODUCTION

Biometric is defined as the characteristics of human which is unique and can be measured for uniqueness or digital authentication purpose. It is the combination of three fields that is image processing, pattern recognition and computer storage system. While defining a characteristic for any biometric system some of the important points that should be noted, every individual should possess defined characteristics like face, iris, palm prints etc, and it should be static in nature. Face recognition and pose estimation is the key area for research in biometric, Face recognition technology (FRT) is embedded with head pose estimation for recognition of individuals with enhanced accuracy and effectiveness [1]. For estimating head pose having different expression and illumination, Parametric Linear Subspace (PLS) method is used. The system combines both verification and identification of images. Gabor wavelets are used

Corresponding Author : Md Rashid Mahmood, ECE department, Guru Nanak Institutions Technical Campus, Hyderabad, India; e-mail : er.mrashid@gmail.com

How to cite this article : Mahmood, Md., R., Mohan, A., Raja, R., Gupta, A. (2020). Parametric Linear Subspace Method for Face Recognition and Pose Estimation. *SAMRIDDHI : A Journal of Physical Sciences, Engineering and Technology, Volume 12, Special Issue (3), 150-156.*

Source of support : Nil

Conflict of interest : None

in this system for efficient and effective detection of land marks on the face (nose, eyes, mouth, etc.) [2-3].

Face recognition and pose estimation is a key research area in field of biometrics, computer vision, and pattern recognition. In the word of Graphics and machine learning also Face Recognition Technology (FRT) is increasingly becoming popular

Vehicle Detection on Sanctuaries Using Spatially Distributed Convolutional Neural Network

Ramakant Chandrakar^{*1}, Rohit Raja², Rohit Miri³, S R Tandan⁴

¹ Research Scholar, Department of CSE, Dr. C. V. RAMAN University, Bilaspur, India; e-mail : ramakant.chandrakar42@gmail.com

² IT Department, GGV Central University, Bilaspur Central University, Bilaspur, India; e-mail : drrohitraja1982@gmail.com

^{3,4} Department of CSE, Dr. C. V. RAMAN University Bilaspur, India; e-mail : rohitmiri@gmail.com, srtandan26@gmail.com

ABSTRACT

Nowadays, detection of vehicles from images captured using web camera is becoming an important focus in the research field of image processing. A common issue inside a wildlife sanctuary is that the possibility of vehicle moving in wrong way or getting into some problem is high. This work proposes an algorithm for detection of vehicles using CNN (Convolutional neural network) on the basis of SDP (spatially distributed pooling). Finite length feature vector is developed by sampling various sized behavioural pattern using SDP. Thus effect of detection can be improved by avoiding distortion of different sized images. Also, Normed slope (NS) method is proposed with more number of threshold as an algorithm for image pre-processing. Using NS, retains the object edge which might be disturbed in the infrastructure. Computational cost for extraction of candidate objects is lesser as only a limited candidate windows are generated. Results of experiments reveal that the SDP based CNN can work well on multiple sized input images thereby improving the effect of detection.

Keywords: Web Camera, Detecting vehicles, Spatially Distributed Pooling, convolution neural network.

SAMRIDDHI : A Journal of Physical Sciences, Engineering and Technology, (2020); DOI : 10.18090/samriddhi.v12iS3.24

INTRODUCTION

Web cameras are being developed nowadays as light-weight, low cost devices. The images captured using such devices can be used to obtain data about remote sensing including, observation of traffic, observing agricultural land cover [1-6], archaeology [7-9], meteorology [10], monitoring volcanic regions [11] and monitoring forest fires [12]. In dense forest because of the absence of mobile network for communication, web cameras are mainly utilised for taking rescue measure. In a sanctuary, the issue of getting missed or being trapped is a major problem concerning the lives of individuals involved on it and hence the providing security is critical. In addition to this, there may arise issues like getting into improper way, malfunctioning of vehicle etc. The seriousness of such issues is at peak during night as well as morning because the possibility of getting injured by deadly animals is

Corresponding Author : Ramakant Chandrakar, Research Scholar, Department of CSE, Dr. C. V. RAMAN University, Bilaspur, India; e-mail : ramakant.chandrakar42@gmail.com

How to cite this article : Chandrakar, R., Raja, R., Miri, R., Tandan, S.R. (2020). Vehicle Detection on Sanctuaries Using Spatially Distributed Convolutional Neural Network. *SAMRIDDHI : A Journal of Physical Sciences, Engineering and Technology, Volume 12, Special Issue (3), 116-121.*

Source of support : Nil

Conflict of interest : None

more at these times. Due to the absence for the medium of communication it becomes impossible to being helped at such situations. It is more difficult for monitoring every vehicle entering into a sanctuary as the number of visitors are high and it is critical for monitoring them in the mid of forests.

©The Author(s). 2020 Open Access This article is distributed under the term of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and non-commercial reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if change were made. The Creative Commons Public Domain Dedication waiver (<http://creativecommons.org/publicdomain/zero/1.0/>) applies to the data made available in this article, unless otherwise stated.

Lung Cancer Detection Using Deep Convolutional Neural Networks

^{1*}Laxmikant Tiwari, ²Rohit Raja, ³Vineet Awasthi, ⁴Rohit Miri, ⁵Akanksha Gupta

¹Research Scholar, Department of CSIT, Dr. C. V. RAMAN University, Bilaspur, India,
laxmikant.tiwari@gmail.com

²Associate Professor, Department of IT, GGV Central University, Bilaspur, India,
drrohitraja1982@gmail.com

³Professor, Department of Information Technology, Dr. C. V. RAMAN University Bilaspur, India

⁴Professor, Department of Information Technology, Dr. C. V. RAMAN University Bilaspur, India,
rohitmiri@gmail.com

⁵Assistant Professor, Department of Information Technology, GGV Central University, Bilaspur Central University India.

Abstract:

Lung Cancer growth is one of the significant reasons for disease-related Deaths because of its forceful nature and postponed identifications at cutting edge phases. Primary identification of lung disease is substantial for the endurance of a person and remains a critical testing issue. Most chest radiographs (X-beam) and recorded tomography (CT) scans are used at first to assess the hazardous knobs; in either event, the possible existence of polite keys prompts inappropriate decisions. At the initial phases, the nice and fragile knobs are next to each other. Here, a new, profound learning-based paradigm with different approaches is suggested to examine the dangerous knobs exactly. Owing to ongoing achievements in in-depth convolutionary neural systems (CNN) in picture analysis, we used two extreme three-dimensional (3D) altered blended relation arrangement (CMixNet) constructs independently for lung knob exploration and characterization. Knob recognition was performed via faster R-CNN on effectively taking highlights from CMixNet and U-Net like encoder-decoder engineering. Characterization of the knobs was accomplished by an orientation boosting machine (GBM) on the outlines of the intended 3D CMixNet layout. To reject false positive outcomes and misdiagnosis due to various mistake types, an acceptable urge was acted on clinical side effects and clinical pathogenesis. Through the network of things (IoT) approach and electro-clinical engineering, remote body popular frameworks (WBANs) provide reliable patient management, helping to decide endless diseases — particularly metastatic sicknesses. The deep neural network for identification and classification of knobs, related to clinical components, begins to reduce disorder, and false optimism (FP) contributes to discovering the initial step of lung disease. The suggested system was tested as affectability (94%) and explicitness (91%) on LIDC-IDRI datasets, and better findings were obtained in comparison to current techniques. In this article, we analyze the consistency of a deep learning technique to diagnose lung disease on clinical image analysis problems. Convolutionary neural systems (CNNs) have become popular within example recognition and PC vision testing territories as a function of their encouraging impact on substantial level representations.


Keywords: Computerized tomography; Deep Neural Network; Image Recognition; lung Cancer; clinical imaging

1. Introduction:

As specified by the World Health Organization (WHO) the lung malignancy is named a non-communicable illness, and it is the fifth reason for death (related to windpipe and bronchus disease) among every single imaginable reason assessed on the planet ("GHO |," n.d.). Lung Cancer growth is the most widely recognized kind of disease influencing the lives of numerous individuals worldwide for a very long while, which reasons for death are evaluated for almost 1.59 million individuals every year. As per Siegel (Siegel et al., 2014) is normal for 2017 that lung Cancer growth, among all tumors, presents the most noteworthy occurrence for new cases (222,500 cases) thinking about people as one gathering, and the most elevated mortality (155,870 Deaths) for the two sexual orientations in



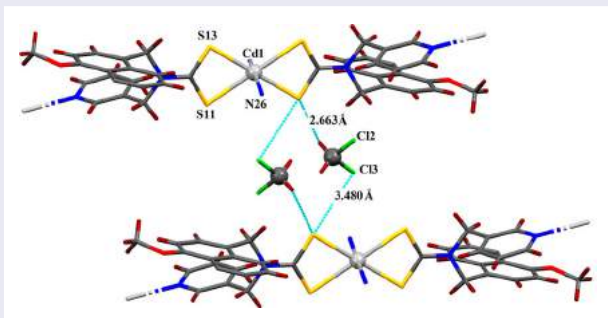
Co-operative influence of co-crystallized solvent in sustaining supramolecular architectures of Zn(II)/Cd(II) homoleptic pyridyl functionalized dithiocarbamates complexes *via* non-covalent interactions

Vinod Kumar ^a and Suryabhan Singh ^b

^aDepartment of Chemistry, B.S.A. College, Mathura, India; ^bDepartment of Chemistry, Guru Ghasidas Vishwavidyalaya, Bilaspur, India

ABSTRACT

Homoleptic pyridyl-3(N) functionalized dithiocarbamate complexes namely, $[Zn(L1)_2]_2$ **1** ($L1 = (C_5H_4NCH_2)_2NCS_2$) and $[Cd(L2)_2]_\infty$ **2** ($L2 = CH_3C_6H_4CH_2NCS_2CH_2C_4H_3O$) have been synthesized and characterized by elemental analysis and different spectroscopic techniques. Single crystal structural studies reveal that compound **1** is binuclear while, **2** is polymeric structure. The presence of weak interactions between complexes and solvent molecules in the lattice results in one-dimensional and three-dimensional supramolecular structures of **1** and **2** respectively. In the case of complex **2**, a rare and significant S...Cl chalcogen bonding interaction was found, which involve the solvent molecule in chain motifs. Complex **2** is a unique example of homoleptic metal dithiocarbamate complexes exhibiting such types of interactions. Complexes showed emission in the solid state. The TGA studies of complexes revealed that complex **1** would be a good precursor for the preparation of ZnS thin film *via* metal organic chemical vapor deposition (MOCVD).




ARTICLE HISTORY

Received 13 June 2021
Accepted 1 November 2021

KEYWORDS

Dithiocarbamate; chalcogen bonds; photoluminescence; polymeric complex

CONTACT Vinod Kumar  vchem20@gmail.com  Department of Chemistry, B.S.A. College, Mathura 281004, India

 Supplemental data for this article can be accessed here. <https://doi.org/10.1080/17415993.2021.2002863>

Design of a Novel Ensemble Model of Classification Technique for Gene-Expression Data of Lung Cancer with Modified Genetic Algorithm

Prem Kumar Chandrakar^{1,*}, Akhilesh Kumar Shrivastava², Neelam Sahu³

¹Department of Computer Science, Mahant Laxminarayan Das College, Raipur (C.G.) India.

²Department of Computer Science and Information Technology, Guru Ghasidas Vishwavidyalaya, Bilaspur. India.

³Department of Information Technology and Computer Science, Dr. C.V. Raman University, Kota, Bilaspur. India.

Abstract

INTRODUCTION: Gene expression levels are important for identifying and diagnosing diseases like cancer. Gene expression microarray information contains a high extent feature set, which minimizes the performance and the accuracy of classifiers.

OBJECTIVES: This paper proposes a Modified Genetic Algorithm (MGA) that is based on Classifier Subset Evaluators – Genetic Search (Eval-CSE_GS) for selecting the relevant feature subsets. The MGA feature selection procedure is applied to microarray information for cancer patients that minimize a high dimension feature subset into low dimension feature subsets.

METHODS: The various data mining methods for classifying the various kinds of cancer disease patients are presented. The proposed model refers to an ensemble model (PEM) for the organization of cancer disease by reducing the feature subsets, which results show improvements in the success rate.

RESULTS: The proposed ensemble model obtains the accuracy of 94.58%, 96.56% and 97.04% for PEM-1 to PEM-3, respectively.

CONCLUSION: Our proposed MGA-PEM model gives satisfactory results for cancer identification and classification.

Received on 27 August 2020; accepted on 12 December 2020; published on 08 January 2021

Keywords: Gene Expression, Modified Genetic Algorithm (MGA), Ensemble, Proposed Ensemble Model (PEM), Microarray, Lung Cancer.

Copyright © 2021 Prem Kumar *et al.*, licensed to EAI. This is an open access article distributed under the terms of the Creative Commons Attribution license (<http://creativecommons.org/licenses/by/3.0/>), which permits unlimited use, distribution and reproduction in any medium so long as the original work is properly cited.

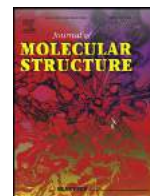
doi:10.4108/eai.8-1-2021.167845

1. Introduction

Gene-expression patterns are attributes of disorder diagnosis, which can be applied to accurately classify cancer. Nowadays, many data mining and classification strategies like Naive Bayes and J-48 are being developed in the research community, in which most of the methods are applied to cancer disorder data and its organization [1, 2]. This supported organic phenomenon amounts from microarray information and gene-disease relationships may be detected using machine-learning algorithms and owing to the high dimensionality of microarray information data sets,

which is often challenged with over-fitting, poor performance, and low potency. Given these challenges, there are some discrimination methods for the classification of tumors proposed by Dudoit [3] through the use of high-density DNA sequences and oligonucleotides. Feature choice [4], ensemble call trees [5], and ensemble neural networks [6] also appear to be effective and possible solutions. Although many researchers have explored cancer classification, few of them have centered on the combinatorial ensemble methodology with a support vector machine or are inconclusive in terms of the classifiers' performance. This paper proposes a Modified Genetic-Algorithm (MGA) and Projected Ensemble Model (PEM) as a learning algorithmic rule that remarkably improves the accuracy and strength of

*Corresponding author. Email: prem.k.chandrakar@gmail.com



Evolution of metal-thiocarboxylate chemistry in 21st century

Suryabhan Singh

Department of Chemistry, Guru Ghasidas Vishwavidyalaya, Bilaspur-495009, India



ARTICLE INFO

Article history:

Received 7 December 2020

Revised 2 February 2021

Accepted 13 February 2021

Available online 28 February 2021

Keywords:

Thiocarboxylate
metal complexes
Coordination chemistry
single source precursor
metal sulfide

ABSTRACT

Metal-thiocarboxylate complexes have interesting features in coordination chemistry as well as in material applications along with some role in the biological system. Which are glint by 160 research papers and more than 350 metal thiocarboxylate compounds have been reported by the researcher in this century. Most of the thiocarboxylate complexes, structurally characterized by single-crystal X-ray diffractometer and their applications have been studied in materials and others. The present review surveys the syntheses, structures, and applications of metal thiocarboxylates.

© 2021 Elsevier B.V. All rights reserved.

1. Introduction

For many years metal complexes of thiocarboxylate ligands have stimulated considerable interest concerning their synthetic and structural aspects. Renewed interest in the chemistry of such complexes can be witnessed in recent years due to the recognition of novel structural features [1-11] and the involvement of these in some biological processes [12-26]. Many such complexes have also been found to be excellent precursors for various types of sulfide materials with interesting optoelectronic properties [9,11,27,28]. Metal sulfides in the forms of powders, thin films, and nanoclusters have generated a great deal of scientific and technological interest for different reasons. Several transition metal sulfides exhibit interesting properties like semiconductivity, luminescence, photoconductivity, chemical sensing, catalysis, superconductivity, etc [29-35]. The soft synthetic route (Chimie douce) provided by these complexes have added further interest in their applications [36-38]. The presence of C(O)S unit in thiocarboxylates imparts the capability to bind with all types of metal ions; hard, soft, and borderline and also to stabilize unusual coordination numbers and geometries [1,7]. Thiocarboxylate complexes of a few metals have recently been reported to undergo thiocarboxylic anhydride elimination leading to the formation of metal sulfides [39,40].

In this review, we have summarized recent (last two decades) progress in studies of the syntheses, structures, and applications of metal complexes bearing thiocarboxylate ligands (Table 1). The following summary is classified according to the group in the periodic table.

2. Syntheses and types of thiocarboxylic acid

Several synthetic methods have been employed to prepare various thiocarboxylic acids. A brief overview of different procedures is given below. Schiff et al prepared thioacetic acid by the treatment of phosphorus pentasulfide on acetic acid [41]. The same compound was also prepared by Clarke et al by passing H₂S gas into acetic anhydride in presence of a small quantity of acetyl chloride [42]. Budzikiewicz et al have synthesized different types of thiocarboxylic acids containing heterocyclic ring (thiophene, pyridine, and furan) by applying different methods (either by hydrolysis of thioacid amide or reaction of carbonyl chloride with KSH) [43]. Hildebrand et al synthesized pyridine-2,6-bis(thiocarboxylic) acid by reaction of pyridine-2,6-dicarbonyl chloride and saturated pyridine with hydrogen disulfide followed by acidification [44]. Recently thiophene-2-thiocarboxylic was synthesized by the reaction of 2-thiophenecarbonyl chloride and thioacetamide in the presence of KOH (Scheme 1) [45].

The most common method, however, involves the reaction of an acid chloride with potassium hydrogen sulfide, followed by its acidification (Scheme 2) [44,46].

3. Metal thiocarboxylate complexes

3.1. Alkali and Alkaline earth metal complexes

In the 21st century complexes containing thiocarboxylate, ligands are mostly derived either from transition or main group metals. There are no further reports on thiocarboxylate complexes of group 1 and 2 metals after the complexes reported by Kato et al. [1]. Lack of new report, possibly less structural variability, and ap-

E-mail addresses: sbs.bhu@gmail.com, suryabhan.27@ggu.ac.in



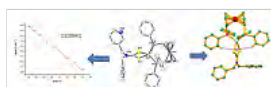
(/)

home (/) > Articles

Synthesis, Crystal Structure, and Properties of Heteroleptic Cu (I) dithiocarbamate complex containing diphenyl phosphinoferrocene (dppf)

(being prepared for publication)

Suryabhan Singh



(/upload/iblock/fff/KUmar.png)

ZhSKh, v. 62, No. 11, 2021, p. 0
(/journal/81141/#art82655)

- Received: April, 20, 2021
- Revised: June, 17, 2021
- Accepted for publication: June, 17, 2021
- DOI 10.26902 / JSC_id82655
- Views: 63

© Kumar V. ¹ ✉ (mailto:vchem20@gmail.com), Singh S. ²¹ Department of Chemistry, BSA College, Mathura, India² Department of Chemistry, Guru Ghasidas Vishwavidyalaya, Bilaspur, India

A new copper (I) heteroleptic pyridyl functionalized dithiocarbamate (dtc) complex, [Cu (L) ₂ dppf] · 2H₂O · MeOH, (1) (where L = N-benzyl-N-methylpyridyl-dtc and dppf = diphenyl phosphinoferrocene), has been synthesized from the reaction of [Cu₂(m-Br)₂(κ²-P, P-dppf)₂] and dithiocarbamate ligand (L). The synthesized complex has been characterized by elemental analysis, spectroscopy techniques (IR, ¹H, ¹³C, ³¹P NMR, and UV-Vis.), and single-crystal X-ray crystallography. In this heteroleptic complex, the Cu atom forms distorted tetrahedral coordination geometry. The supramolecular architecture in the complex has been sustained in the solid phase by, C–H...O and CH...π (chelate = CuS₂C) interactions. The emission spectrum of the complex has been studied in DCM solution. The charge-transfer excited state is quenched due to intramolecular energy transfer from the {Cu (S, S) (P, P)} moiety to the ferrocene therefore dppf-based complex shows no detectable emission at room temperature. This complex is weakly conducting and exhibit semiconductor behavior at room temperature.

Keywords: Heteroleptic / Cu (I) dithiocarbamate complex / Crystal structure / diphenyl phosphinoferrocene (dppf) / Semiconductor

	2019	2020
Impact Factor	0.745	↑ 1.071
Publication period	6 months	

Dear authors and subscribers of the "Journal of Structural Chemistry", the archive of ZhSKh until 2017 is freely available ("Numbers" (<https://jsc.niic.nsc.ru/numbers/>)).

Place an advertisement in ZhSH (/docs/news/ПП%20и%20письмо.pdf)



(/docs/news/2020-10-16_Макет_НГУ.pdf)

(as advertising)

Special issues

ЖСХ, т.55, №8, 2014 (/journal/40/)
Самоорганизация молекулярных и супрамолекулярных структур (/journal/40/)



Reviews (/category/327/)

The most interesting

Articles most often viewed by visitors

Study of the layered mineral vallerite by XPS methods ... (1287)

Composition and structure of hydrates of molecules CH₃COOH ... (1025)

Crystal and molecular structure of 2,4-diphenyl-3-(3-phenylpropin-2-yl)-3-azabicyclo [3.3.1] nonan-9-ol (868)



Differential evolution based radial basis function neural network model for reference evapotranspiration estimation

Babita Majhi¹ · Diwakar Naidu^{1,2}

Received: 2 April 2020 / Accepted: 22 December 2020

© The Author(s) 2021

Abstract

The present study is an effort to examine the capability of a differential evolution based radial basis function neural network (RBFDE) to model weekly reference evapotranspiration (ET_0) as a function of climatic parameters in different agro-climatic zones (ACZs) of a moist sub-humid region in East-Central India. The ET_0 computed using the empirical equation of Penman–Monteith suggested by the Food and Agricultural Organization (FAO56-PM) is considered as a target variable for investigation. The performance of the proposed RBFDE model is compared with particle swarm optimization based radial basis function (RBFPSO), radial basis function neural network (RBFNN), multilayer artificial neural network (MLANN) models and conventional empirical equations of Hargreaves, Turc, Open-Pan, and Blaney-Criddle. Weekly ET_0 estimates that are obtained using RBFDE, RBFPSO, and RBFNN and MLANN are observed to be more consistent than equivalent empirical methods. For a critical analysis of simulation results, mean absolute percentage error (MAPE), root means square error (RMSE), determination coefficient (R^2) and Nash–Sutcliffe efficiency factor (NSE) is computed. Low MAPE and RMSE values along with higher R^2 and NSE close to 1, obtained with soft computing models exhibit that, soft computing models produce better estimates of ET_0 than empirical methods. Among the soft computing models, RBFDE provides improved results as compared to RBFPSO, RBFNN, and MLANN models. This method can be extended for ET_0 estimation in other ACZs.

Keywords ET_0 estimation · FAO56-PM · RBFNN · RBFDE · RBFPSO · MLANN · Empirical methods

1 Introduction

In response to atmospheric demand, soil surface evaporation and transpiration from plant occurs simultaneously in a cropping field and is termed as evapotranspiration (ET) in a combined manner [1]. Approximately two-thirds of the total precipitation is consumed by the atmosphere in the form of ET [2]. Therefore, ET is considered one of the most important water balance components for the determination of crop water requirement, length of the crop growing season, and associated agro-climatic studies. Hence, accurate measurement or estimation of ET is

essential for the planning and effective implementation of irrigation and water management practices for practical applications. Accurate measurement of ET by volumetric and gravimetric lysimeter is practically very difficult because various factors affect the ET process, which includes climatic parameters, crop characteristics, soil properties, and management practices. Therefore, consumptive use of water from a uniformly distributed grass reference crop under nonlimiting conditions is estimated for practical purposes and termed as ET_0 [3]. In general, ET_0 is computed employing empirical equations as climatic parameters being the only factor affecting the ET process.

✉ Diwakar Naidu, dnaidu1971@gmail.com | ¹Department of CSIT, Guru Ghasidas Vishwavidyalaya, Central University, Bilaspur 495009, India. ²BRSM College of Agricultural Engineering and Technology and Research Station, Indira Gandhi Krishi Vishwavidyalaya, Mungeli, India.



Solar radiation, temperature, humidity, wind speed, and sunshine are the most influential climatic factors which contribute to the ET process [1]. Precise estimation of ET_0 is vital for the efficient utilization of available water resources for agricultural purposes.

Several physicals, empirical equations based on radiation, temperature, mass transfer, and water budget methods have been derived in the past to determine ET_0 with different input combinations of meteorological parameters. Among these empirical methods, the Penman–Monteith equation is recommended by the Food and Agricultural Organization for ET_0 estimation (FAO56-PM) because of its preciseness [1]. FAO56-PM equation requires meteorological parameters such as maximum and minimum temperatures, relative humidity, sunshine hours, wind speed, and solar radiation to determine ET_0 . In developing countries, like India, it is invariably very difficult to obtain long term meteorological parameters to compute ET_0 using the FAO56-PM model [4]. Therefore, other empirical models like Hargreaves [5], Turc [6], Open Pan [7], Blaney-Criddle [7], and Christianson [8], etc., are also in use. These empirical equations involve fewer complex variables as compared to FAO56-PM to compute ET_0 . However, ET_0 estimates obtained using these models are not comparable with FAO56-PM as these methods yield more errors and hence their practical applications become limited [9].

To address this issue, in recent decades, researchers have successfully demonstrated the application of a variety of computational intelligence based conventional and hybrid soft computing techniques for modeling extremely complex and non-linear relationship between climatic factors and ET_0 [10–19]. Improved predictions of FAO56-PM ET_0 are obtained by Wen et al. [20] using a support vector machine (SVM) as compared to the artificial neural network (ANN) and empirical methods in extremely arid regions of China. Partal [21] has developed a hybrid model combining wavelet transformation and radial basis function neural network (W-RBF) that outperformed conventional RBF, wavelet-multi-linear regression (W-MLR) and empirical methods of Hargreaves and Turc for daily ET_0 estimation with improved accuracy. Kisi and Demir [22] have evaluated the potential of multi-layer perceptron (MLP) with six different weight update algorithms for modeling ET_0 and found MLP with the Levenberg-Marquard algorithm produced a better estimate of ET_0 . In a recent study, Dou and Yang [23] have recommended hybrid extreme learning machine (ELM) and adaptive neuro-fuzzy inference system (ANFIS) based models that are more robust and flexible in comparison to traditional ELM and ANFIS. Adamala [24] has reported improved generalized performance of wavelet neural network (WNN) and ANN model for estimation of ET_0 as compared to linear regression (LR),

wavelet regression (WR) and Hargreaves (HG) methods for the studied locations in different agro-ecological regions of India. Sanikhani et al. [25] have applied several artificial intelligence models including multi-layer perceptron (MLP), generalized regression neural network (GRNN), integrated ANFIS systems with grid partitioning (ANFIS-GP) and subtractive clustering (ANFIS-SC), radial basis neural network (RBNN) and GEP for modeling ET_0 in a cross-station scenario for different locations in Turkey and demonstrated that AI-based models performed better than the empirical equation of Hargreaves-Samani (HS) and its calibrated version (CHS).

It is also observed from the literature review that researchers have successfully implemented various types of hybrid soft computing models combining conventional neural networks along with evolutionary computing algorithms for estimation of ET_0 . Application of nature-inspired algorithms such as genetic algorithm (GA), particle swarm optimization (PSO), artificial bee colony (ABC), etc., in combination with conventional neural networks like ANN and RBNN are investigated in some research publications for ET_0 estimation [26–30]. A study conducted by Feng et al. [31] for estimating FAO56-PM ET_0 in a humid region of Southwest China reveals that ELM and ANN optimized by genetic algorithm (GANN) has resulted in better ET_0 estimates than WNN and empirical approaches of Hargreaves, Makkink, Priestley–Taylor and Ritchie models. Gocić et al. [32] have analyzed the potential of genetic programming (GP), support-vector machine-firefly algorithm (SVM-FFA), ANN, and SVM-Wavelet soft computing approaches and found SVM-Wavelet resulted in improved FAO56-PM ET_0 estimates in Serbia. Mehdizadeh et al. [33] have evaluated the performance of gene expression programming (GEP) and MARS along with two SVM based hybrid models, SVM-Polynomial and SVM-RBF for estimation of monthly mean ET_0 and reported SVM-RBF and MARS outperformed GEP and SVM-Poly and also performed better than 16 other empirical equations considered for comparison. However, Mattar and Alazba [34] have confirmed that the GEP model performed better than the conventional multilinear regression (MLR) approach in Egypt. Most of the soft computing models discussed above are developed under a given scenario in terms of study location, the combination of available input climatic parameters, time scale and duration of climatic data, model structure, learning parameters, and an optimization algorithm, etc. Therefore, practically it becomes very difficult to employ these models in a new location without proper calibration and validation of the model parameters.

To examine the potential of an evolutionary optimized soft computing technique, RBFNN in combination with the differential evolution algorithm (RBFDE) is introduced here for the estimation of ET_0 under three different ACZs in the

Chhattisgarh region of East-Central India. Differential evolution (DE) is considered because it is a simple algorithm in comparison to GA which requires intensive calculations. Due to its simplicity, DE is used in various applications [35–37]. Technical analysis of DE parameters, hybridization of DE with other soft computing techniques, and its practical applications have been discussed by Das et al. [38]. Different variants over state-of-the-art DE have also been presented in the literature. Among these, Hui and Suguntham [39] suggested ensemble and arithmetic recombination-based speciation DE for multimodal optimization of common benchmark problems. Ramdas et al. [40] developed a reconstructed mutation strategy for DE and applied the same with multilevel image thresholding for improved weather radar image segmentation [41]. A DE variant with multi-donor mutation strategy and annealing-based local search has been developed by Ghosh et al. [42] for optimization of Lennard-Jones potential function-based molecular clustering. The effect of DE-based constraint handling techniques has been evaluated by Biswas et al. [43] for the optimization of power flow systems. One of the authors of this investigation has also been engaged in DE based training of adaptive autoregressive moving average (ARMA) model for exchange rate forecasting [44] and development of a hybrid system using functional link artificial neural network (FLANN) and DE for Odia handwritten numeral recognition [45]. The proposed evolutionary optimized hybrid structure of RBFDE is developed and used for the first time to model FAO56-PM ET_0 , and therefore it may be considered as a novel scientific approach for such application. Conventional soft computing techniques like MLANN, RBFNN along with empirical methods of Hargreaves, Turc, Open Pan, and Blaney-Criddle are considered for comparison purposes. Results obtained with RBFDE is also compared with RBFPSO under similar condition. This paper is organized into different sections. Section 1 introduces the problem formulation, literature reviews, and motivation behind the investigation. The detailed description of the data sets, soft computing techniques, and empirical methods are described in the Materials and methods of Sect. 2. Simulation results and comparative performance evaluation of different models are outlined in the results and discussion of Sect. 3. The salient findings of the study are summarized in the conclusion section.

2 Materials and methods

2.1 Study area and dataset

This investigation is carried out to model weekly ET_0 using soft computing techniques. Long term weekly meteorological data (2001 to 2019) of maximum temperature

(T_{max}), minimum temperature (T_{min}), bright sunshine hours (BSS), wind speed (WS), morning relative humidity during (RH_1), afternoon relative humidity (RH_2) and weekly cumulative pan evaporation (EP) are collected from Raipur, Jagdalpur and Ambikapur stations located in three distinct ACZs of Chhattisgarh region in central India (Fig. 1). The climate of Chhattisgarh is moist sub-humid in general with an average annual rainfall of 1200–1400 mm and annual ET_0 losses between 1400 and 1600 mm in different ACZs. Data sets are collected from the India Meteorological Department (IMD) (<https://mausam.imd.gov.in/>) certified observatories located in these stations. These surface meteorological observatories follow the World Meteorological Organization (WMO) guidelines for data collection [46]. WMO guidelines for the observational procedure and quality control are adopted uniformly in these surface meteorological observatories while data acquisition, tabulation, and computation. The online data entry system, itself has an inbuilt quality control mechanism to test the errors like data format, duplicate records, and incorrect units of measurement, impossible values, extremes, and outliers.

Descriptive statistics of different meteorological parameters in terms of mean, high, low, range, standard deviation (SD), and coefficient of variation (CV) are also computed to understand data patterns and to ensure the quality check of data (Table 1). To measure the strength and direction of a linear relationship between two variables, correlation coefficient (R) between meteorological parameters (T_{max} , T_{min} , BSS, WS, RH_1 , RH_2 , and EP) with FAO56-PM ET_0 are also computed (Table 1). Weekly totals of ET_0 are computed using the FAO56-PM equation which is considered as the target output for model development [1].

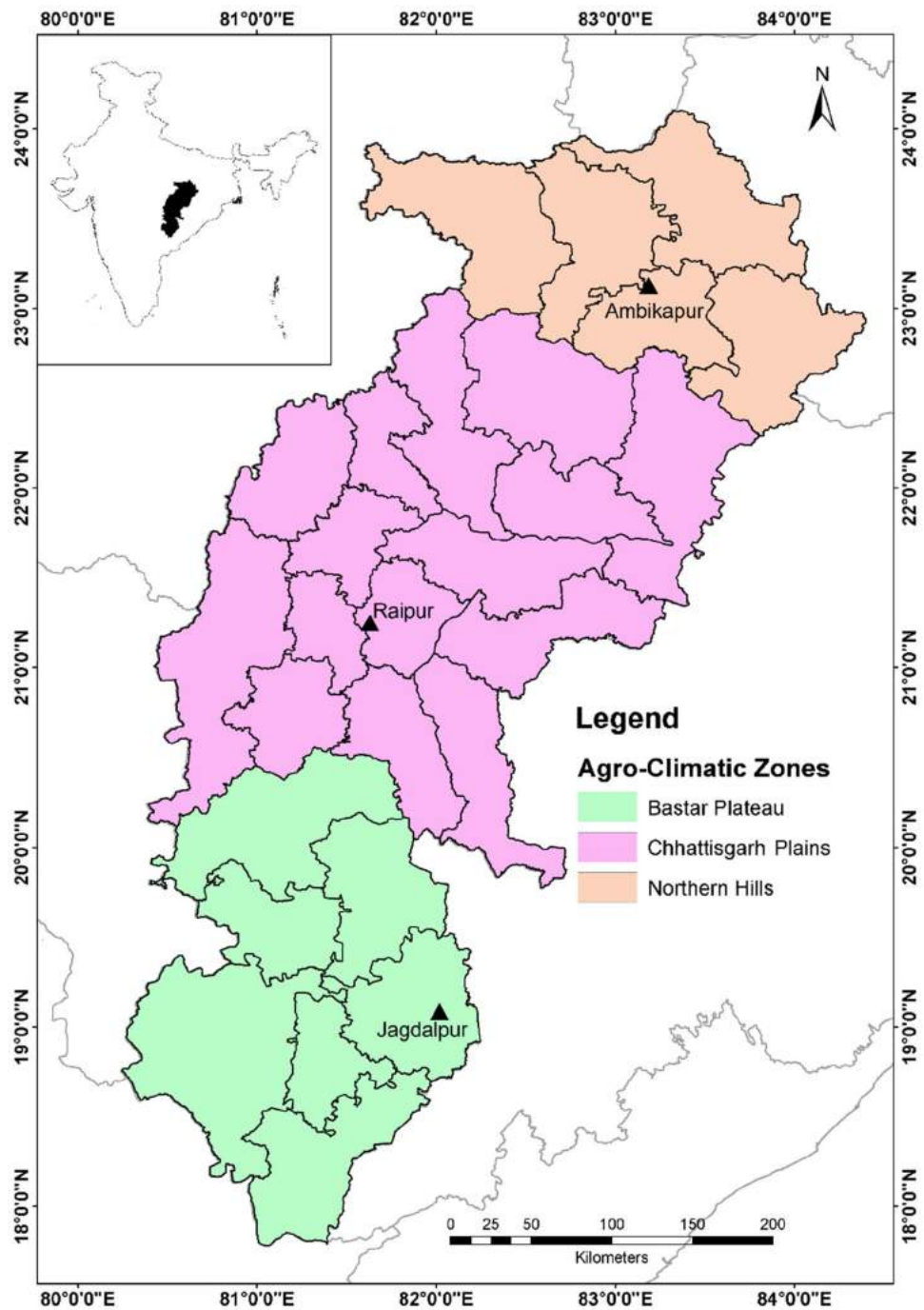
The pattern of different meteorological parameters considered as input variables for model development along with target variable FAO56-PM ET_0 in selected stations is represented as box plot arrangements in Fig. 2. The middle line of the box plot signifies the median value while the upper and lower edges signify 75% and 25% of the data set respectively. The highest and lowest limits of the upper and lower vertical lines indicate the highest and lowest values respectively. The square depicts the simulated mean, and the straight-line shows the observed mean.

2.2 Design of soft computing models

2.2.1 Radial basis function neural network (RBFNN) based estimator

RBFNN is a category of feed-forward neural network with a single hidden layer and an output layer formulated by Broomhead and Lowe [47]. Pictorial representation of the

Fig. 1 Location map of the study area



RBFNN is given in Fig. 3. The processing units termed as neurons in the hidden layer are associated with centers, $c = c_1, c_2, c_3, \dots, c_h$, and their width $\sigma = \sigma_1, \sigma_2, \sigma_3, \dots, \sigma_h$, where h is the number of neurons in the hidden layer. Each neuron in the hidden layer receives the same set of input data ($X = x_1, x_2, x_3, \dots, x_n$). The centers of every hidden neuron have the same dimension as that of the input data, i.e. $c_i \in R^n, X \in R^n$. The output of hidden layer neurons ($\varphi_1, \varphi_2, \varphi_3, \dots, \varphi_h$) are associated with synaptic weights

($w_1, w_2, w_3, \dots, w_h$). Output, θ_i of i th hidden layer neuron is basically a Gaussian function and is represented by:

$$\theta_i(z) = e^{\frac{-z^2}{2\sigma_i^2}} \tag{1}$$

where $z = \|x - c_i\|$, represents the Euclidian distance between input data and the corresponding centers and $\theta_i = \theta(\|x - c_i\|)$. The Gaussian function used in each hidden layer neuron is a category of radial basis function.

Table 1 Descriptive statistics of weekly meteorological parameters (2001–2019) at different locations

Parameters	Mean	High	Low	Range	SD	CV	R
<i>Raipur (21.14°N, 81.38°E, 289 m)</i>							
T_{max} (°C)	32.9	46.0	22.0	24.0	4.9	14.8	0.95
T_{min} (°C)	20.4	31.5	6.6	24.9	5.8	28.6	0.62
BSS (hours)	6.7	11.1	0.0	11.1	2.6	39.5	0.33
WS (Kmph)	4.9	14.9	0.5	14.4	3.1	62.7	0.44
RH_1 (%)	79.5	96.4	25.9	70.5	16.5	20.7	-0.87
RH_2 (%)	43.8	91.0	6.7	84.3	22.2	50.6	-0.47
EP (mm week ⁻¹)	38.0	127.2	10.5	116.7	22.0	58.0	0.97
FAO56-PM ET_0 (mm week ⁻¹)	29.7	68.7	11.0	57.7	12.2	41.1	1.00
<i>Jagdalpur (19.08°N, 82.01°E, 564 m)</i>							
T_{max} (°C)	31.0	42.6	23.3	19.3	4.0	12.8	0.93
T_{min} (°C)	18.1	28.1	4.3	23.8	5.7	31.2	0.47
BSS (hours)	6.2	11.0	0.0	11.0	2.8	44.9	0.43
WS (Kmph)	4.5	11.6	1.0	10.6	2.0	44.3	0.36
RH_1 (%)	86.9	97.7	34.0	63.7	9.4	10.8	-0.83
RH_2 (%)	51.1	95.7	7.4	88.3	20.5	40.1	-0.56
EP (mm week ⁻¹)	29.1	93.4	3.8	89.6	14.6	50.0	0.91
FAO56-PM ET_0 (mm week ⁻¹)	26.8	61.2	13.0	48.2	8.9	33.3	1.00
<i>Ambikapur (23.12°N, 83.20°E, 604 m)</i>							
T_{max} (°C)	30.4	43.6	18.9	24.7	5.12	16.9	0.931
T_{min} (°C)	17.8	28.9	3.1	25.8	6.32	35.5	0.647
BSS (hours)	7.2	11.1	0.2	10.9	2.59	36.1	0.351
WS (Kmph)	3.5	10.4	0.3	10.1	1.79	50.6	0.616
RH_1 (%)	78.8	98.9	25.7	73.2	17.12	21.7	-0.837
RH_2 (%)	46.6	91.6	9.6	82.0	21.62	46.4	-0.452
EP ₀ (mm week ⁻¹)	31.7	92.5	8.8	83.7	16.19	51.0	0.909
FAO56-PM ET_0 (mm week ⁻¹)	26.7	61.8	11.5	50.4	10.52	39.4	1.000

Finally, the response of the RBFNN at the output layer, for a given set of input data is linear in terms of weights and computed using the following expression.

$$y = \sum_{i=1}^h w_i \phi_i \quad (2)$$

Development of the RBFNN for each instant of input data and its corresponding output $\{X, y\}$ is obtained recursively by updating the network parameters $\{w_i, c_i, \sigma_i\}$ to minimize the instantaneous error cost function given as.

$$e = \frac{1}{2} (y^d - y)^2 \quad (3)$$

The weight update rules to optimize the network parameters $\{w_i, c_i, \sigma_i\}$ at time t are given by following equations which are derived using gradient descent algorithm [48].

$$w_i(t+1) = w_i(t) + \eta_1 (y^d - y) \phi_i \quad (4)$$

$$c_{ij}(t+1) = c_{ij}(t) + \frac{\eta_2}{\sigma_i^2} (y^d - y) w_i \phi_i (x_j - c_{ij}) \quad (5)$$

$$\sigma_i(t+1) = \sigma_i(t) + \frac{\eta_3}{\sigma_i^3} (y^d - y) w_i \phi_i z_i^2 \quad (6)$$

where y^d desired output or target value, c_{ij}^{jth} element of the i^{th} center, η_1, η_2, η_3 learning rates for network parameters $\{w_i, c_i, \sigma_i\}$ respectively.

2.2.2 Differential evolution based RBF neural network estimator

Differential evolution (DE) [49, 50] is a simple and efficient global optimization technique based on a heuristic method for minimizing a nonlinear function. Using this efficient heuristic approach a hybrid structure, RBFDE is developed in which total d number of network parameters, represented by a parameter vector, $\vec{x}_i = \{w_i, c_i, \sigma_i\}$, is optimized by the differential evolution algorithm (DE).

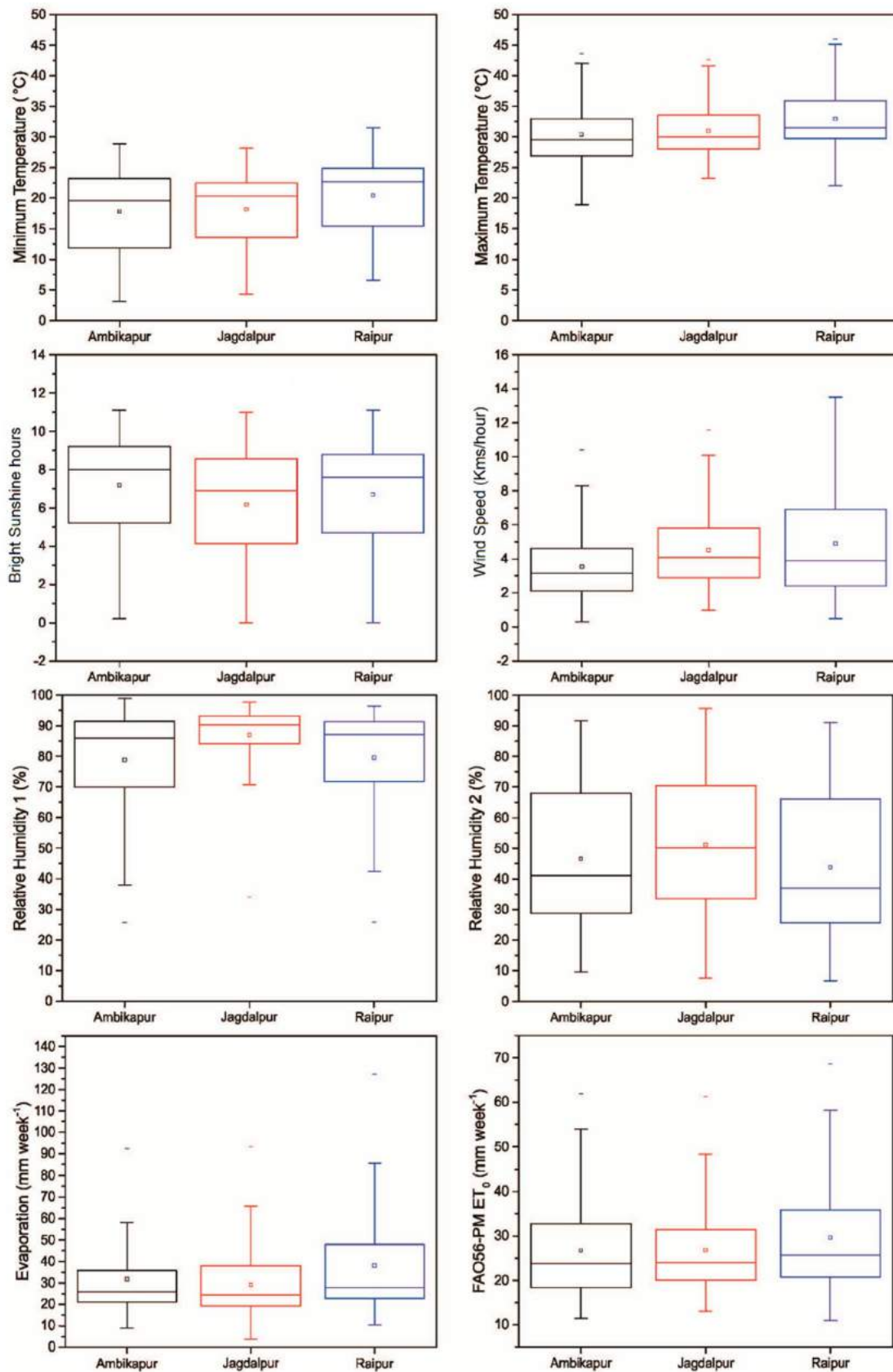


Fig. 2 Box plot of input meteorological parameters and FAO56-PM ET₀ in different stations

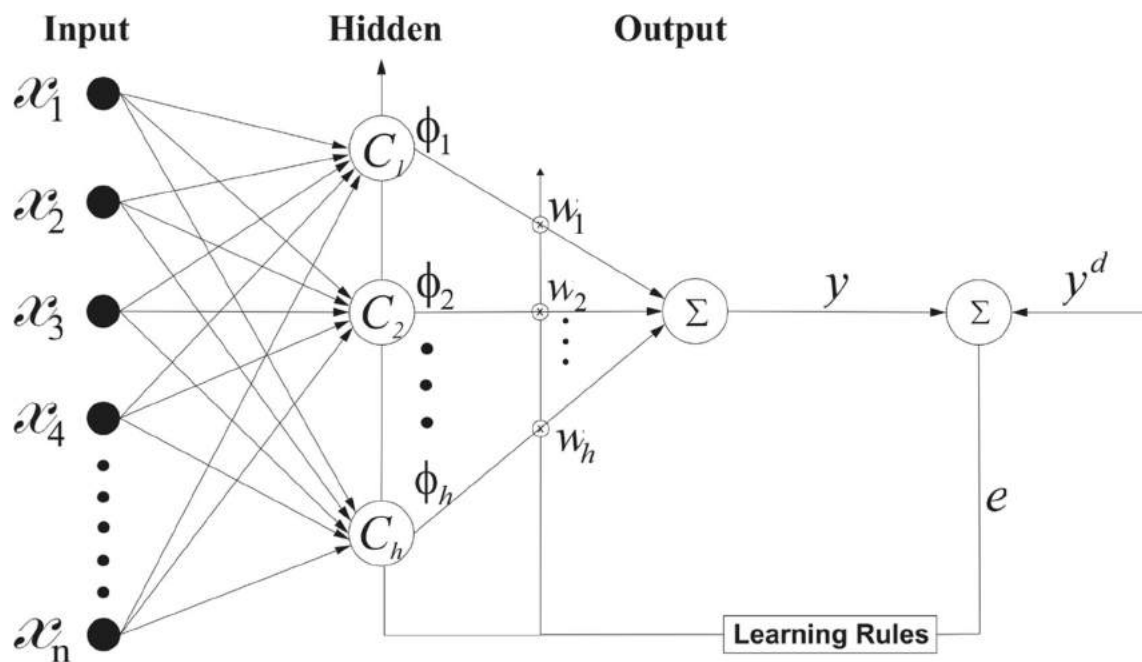


Fig. 3 Block diagram of RBFNN based estimator

DE algorithm involves three basic operations viz., mutation, recombination, and selection. The step-wise procedure for the development of RBFDE is described below.

Step 1 Randomly initialize $i = 1, 2, 3, \dots, NP$ number of target or population vectors, $\vec{x}_{i,G}$ between 0 to 1, where each i th individual of the population vector represents parameters of the RBFDE model. The i th target vector of G th generation, $\vec{x}_{i,G}$ is given as $\vec{x}_{i,G} = \{w_i, c_i, \sigma_i\}$.

Step 2 Repeat step 3 with each target vector $\vec{x}_{i,G}$ for $i = 1, 2, 3, \dots, NP$.

Step 3 a) Give K numbers of input patterns to the RBF network sequentially with each pattern having dimension n .

b) For each one of the K input patterns, obtain corresponding network output using i th target vector $\vec{x}_{i,G}$ as the parameters of the network and compare it with the corresponding desired output to get an error using (3). For K patterns, the K number of error values will be obtained.

c) Calculate $f(\vec{x}_{i,G})$ using (7), where $f(\vec{x}_{i,G})$ represents the fitness function i.e. mean square error (MSE).

$$f(\vec{x}_{i,G}) = \frac{\sum_{j=1}^K e^2}{K} \tag{7}$$

Step 4 Obtain $f_{min}(\vec{x}_{i,G})$ and represent the corresponding $\vec{x}_{i,G}$ as the $\vec{x}_{best,G}$ for G th generation.

Step 5 Choose a scaling factor $F \in [0, 1]$ and a cross over ratio $CR, \in [0, 1]$ and repeat step 6 to step 15 until the desired minimum MSE is obtained.

Step 6 Repeat the steps from 7 to 8 for $i = 1, 2, 3, \dots, NP$ times

Step 7 Randomly choose two indices r_1, r_2 from 1 to NP , such that, $r_1 \neq r_2 \neq i$.

Step 8 Compute the mutant vector $\vec{v}_{i,G}$ for each target vector $\vec{x}_{i,G}$ for G th generation as

$$\vec{v}_{i,G} = \vec{x}_{i,G} + F * (\vec{x}_{r_1,G} - \vec{x}_{r_2,G}) + F * (\vec{x}_{best,G} - \vec{x}_{i,G}). \tag{8}$$

Step 9 Repeat the steps from 10 to 11 for $i = 1, 2, 3, \dots, NP$ times

Step 10 Randomly choose an index r_3 between 1 to d and repeat step 11 for $j=1$ to d , where d is the dimension of the target or population vector.

Step 11 Generate a random number $rand \in [0, 1]$ and compute the trial vector $\vec{U}_{j,i,G}$ by recombination operation, which replaces the previously successful individuals with mutant vector as

$$\vec{U}_{j,i,G} = \begin{cases} \vec{v}_{j,i,G} & \text{if } (rand \leq CR) \text{ or } j = r_3 \\ else & \\ \vec{x}_{j,i,G} & \end{cases} \tag{9}$$

Step 12 For each trial vector $\vec{U}_{j,i,G}, i = 1, 2, 3, \dots, NP$, evaluate $f(U_{i,G})$, which is a mean square error (MSE). (Similar to step 3)

Step 13 Repeat the step14 for $i = 1, 2, 3, \dots, NP$

Step 14 Finally, the next generation of NP number of target/population vector $\vec{x}_{i,G+1}$ is selected based on survival of the fittest criteria as

$$\vec{x}_{i,G+1} = \begin{cases} \vec{U}_{i,G} & \text{if } f(\vec{U}_{i,G}) \leq f(\vec{x}_{i,G}) \\ \text{else} & \\ \vec{x}_{i,G} & \end{cases} \quad (10)$$

Step 15 Obtain $f_{\min}(\vec{x}_{i,G+1})$ and represent it as $\vec{x}_{best,G+1}$ for the next generation.

Step 16 Stop

Pictorial representation of the DE algorithm is shown in Fig. 4.

2.2.3 Particle swarm optimization based RBF neural network estimator

In this approach parameters of the RBFNN model i.e. $\{w_{ij}, c_{ij}, \sigma_j\}$, as described in Sect. 2.2.1, are updated using the PSO algorithm. The PSO [51–53] is a metaheuristics optimization algorithm inspired by the paradigm of swarm intelligence which mimics the social behavior of animals like fish and birds. It is successfully applied to various applications in engineering and science [54–56]. The algorithm uses a fixed number of particles that represent the parameters of RBFNN. Each particle updates its current velocity and position by its own experience called personal best (*p-best*) and by the social experience of the swarm called global best (*g-best*). Steps involved in PSO are briefly described below:

Step 1 Initialize fix number of particles with random position and velocity uniformly distributed over the search space.

Step 2 Evaluate the fitness of each particle according to the objective function

Step 3 Record *pbest* for each particle and *g-best* of the swarm.

Step 4 Update velocity of each particle

Step 5 Update the position of each particle.

Step 6 Update *pbest* and *gbest*

Step 7 Repeat the steps from 2 to 6 until the termination condition is satisfied and stop.

Pictorial representation of the PSO algorithm is shown in Fig. 5.

2.2.4 Multi-layer artificial neural network (MLANN)

MLANN, suggested by Haykin [57] is successfully employed in many applications to solve the regression problem. MLANN architecture considered for this proposed investigation consists of an N-5-1 structure. N represents the number of input features. Optimum results are obtained with 5 neurons in the intermittent hidden layer. Desired ET_0 estimates are obtained at output neurons. Hyperbolic tangent (tanh) is used as an activation function in every processing neuron. The training of the network is done by a conventional back-propagation algorithm which is based on the error-correcting learning rule to update the weights and bias of each neuron in different layers.

2.3 Empirical models

Weekly ET_0 for the study locations is also computed using empirical methods of FAO56-PM, Blaney-Criddle, Open Pan, Turc, and Hargreaves from available meteorological data. A brief description regarding empirical approaches considered in this investigation and the corresponding input meteorological parameter requirement are listed in Table 2. The description regarding different climate based empirical methods considered in this investigation is not included in this paper. More details regarding these empirical approaches can be obtained from basic references [1, 5–7].

2.4 Performance evaluation measures

Comparative analysis of estimated ET_0 obtained with different soft computing models and empirical methods considered for the investigation is carried out by computing performance evaluation measures, namely, mean square percentage error (MAPE), root mean square error (RMSE), determination coefficient (R^2) and efficiency factor (EF) proposed by Nash and Sutcliffe (NSE) [58]. The mathematical expression of different evaluation measures is as follows.

$$MAPE = \frac{1}{n} \sum_{i=1}^n \left| \frac{(Out_{obs} - Out_{est})}{Out_{obs}} \right| \times 100 \quad (11)$$

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (Out_{est} - Out_{obs})^2} \quad (12)$$

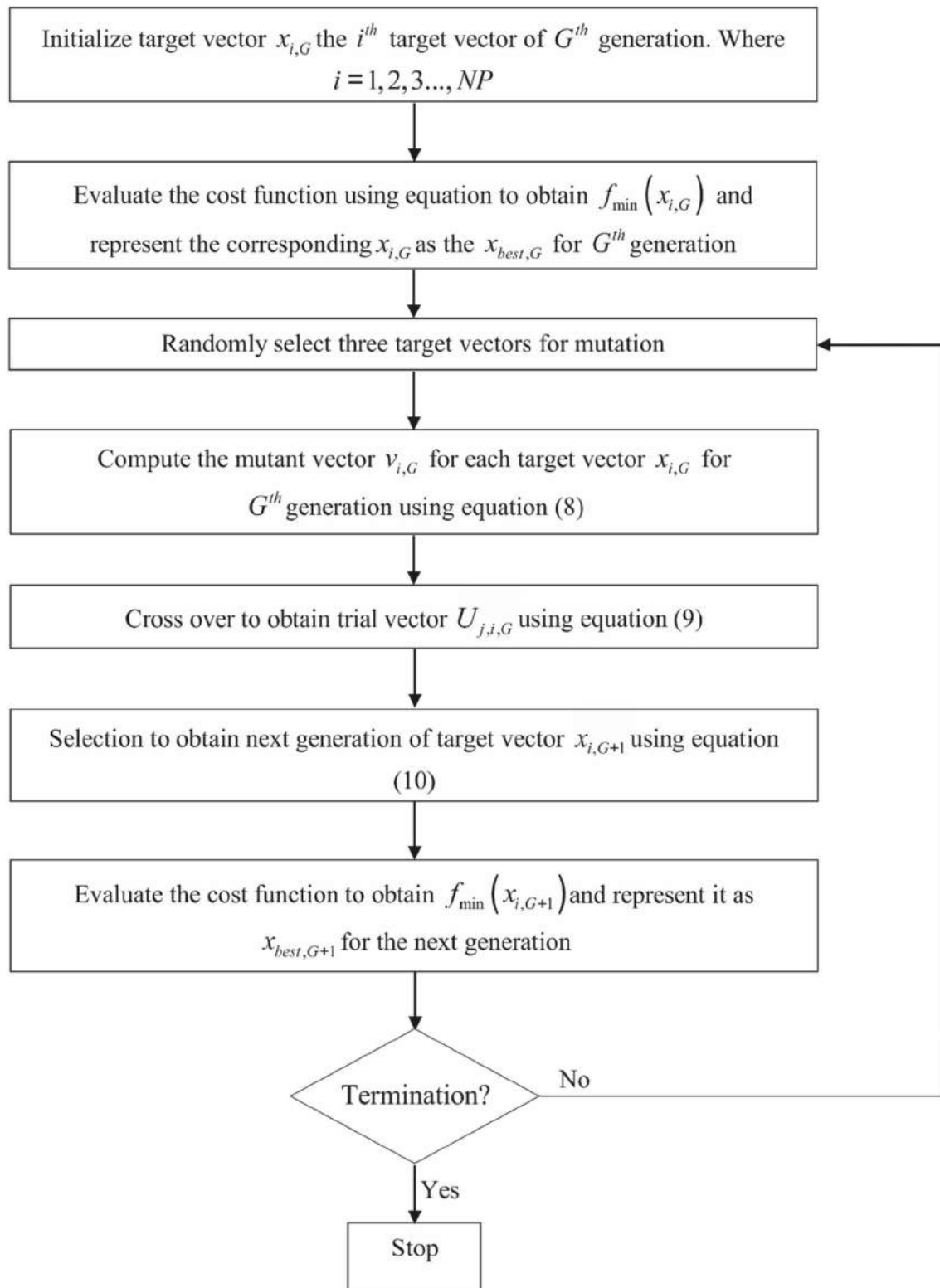


Fig. 4 Flowchart differential evolution algorithm

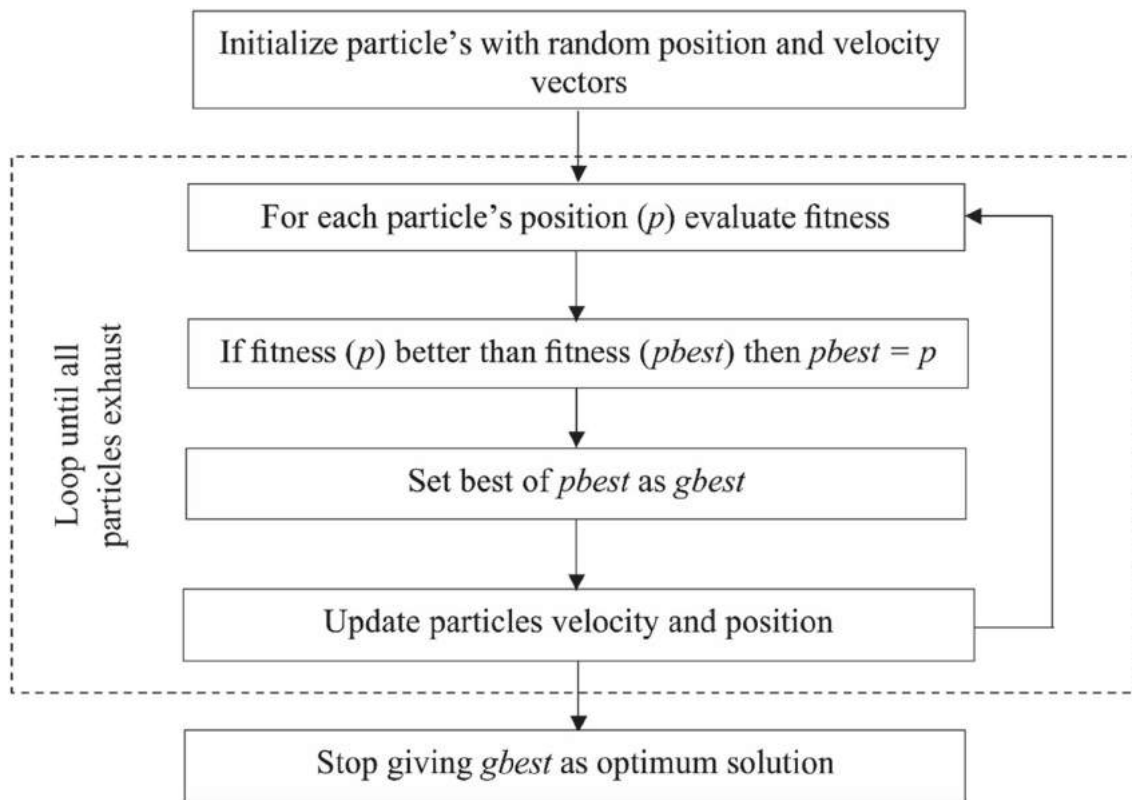


Fig. 5 Flowchart particle swarm optimization algorithm

Table 2 Details of empirical models used to compute ET₀

Empirical methods	Input combination	Empirical equation
Hargreaves [5]	$T_{max} T_{min}$	$ET_0 = 0.0023 \times R_a \times T_d \times 0.5 \times (T_m + 17.8)$
Turc [6]	$T_{max} T_{min} BSS$	$ET_0 = \frac{0.40 \times T_m (R_s + 50)}{(T + 15)}$
Blaney-Criddle [7]	$T_{max} T_{min} RH_1 \& RH_2, WS, BSS$	$ET_0 = a + b \times p(0.46 \times T_m + 8.13)$
Open Pan [7]	$WS, EP, RH_1 \& RH_2$	$ET_0 = k_p \times EP$
FAO56-PM [1]	$T_{max} T_{min} RH_1 \& RH_2, WS, BSS$	$ET_0 = \frac{0.408 \Delta (R_n - G) + \gamma \frac{900}{T + 273} u_2 (e_s - e_a)}{\Delta + \gamma (1 + 0.34 u_2)}$

$$R^2 = \frac{\left(\sum_{i=1}^n (Out_{obs} - \overline{Out_{obs}}) (Out_{est} - \overline{Out_{est}}) \right)^2}{\sum_{i=1}^n (Out_{obs} - \overline{Out_{obs}})^2 \sum_{i=1}^n (Out_{est} - \overline{Out_{est}})^2} \tag{13}$$

$$EF = NSE = 1 - \frac{\sum_{i=1}^n (Out_{est} - Out_{obs})^2}{\sum_{i=1}^n (Out_{obs} - \overline{Out_{obs}})^2} \quad (-\infty \leq EF \leq 1) \tag{14}$$

where Out_{obs} denotes the target and Out_{est} denotes model estimated ET₀ values. n is the number of testing patterns. Low MAPE and RMSE values represent the close agreement between desired and estimated output. Similarly,

R² and EF values close to 1 are also indicators of a higher accuracy level of the model.

3 Results and discussion

The key objective of this investigation is to examine the potential of different evolutionary optimized hybrid (RBFDE, RBFPSO) and conventional (RBFNN, MLANN) soft computing approaches with available climatic features for estimation of ET₀ comparable to FAO56-PM ET₀. Input features combination of different models is decided based on empirical approaches of Hargreaves, Turc, Open Pan, Blaney-Criddle, and FAO56-PM ET₀ listed in the previous section. These soft computing models are categorized

into type I to type V models. Like the Hargreaves method, type I models include only T_{max} and T_{min} as input features, whereas Type II soft computing models include BSS with temperature, which is equivalent to the Turc approach. In type III soft computing models, EP, RH_1 , RH_2 , and WS are considered as input features similar to that of the Open Pan empirical approach. Type IV models include six weather parameters (T_{max} , T_{min} , BSS, WS, and RH_1 and RH_2) equivalent to the Blaney-Criddle empirical approach. Another category of soft computing model termed Type V models are developed using T_{max} , T_{min} , BSS, and WS since these weather parameters exhibit positive correlations with ET_0 . The input feature EP, which is also positively correlated with ET_0 , is not included in type V soft computing models as obtaining EP data is very difficult. Input feature combinations used in different types of soft computing models and their equivalent empirical models are shown in Table 3. Weekly meteorological data of Raipur, Jagdalpur, and Ambikapur from 2001 to 2015 (80%) are used for model calibration or training, whereas the recent 4 years (20%) of the weekly meteorological data from 2016 to 2019 are used for model validation.

Soft computing models RBFDE, RBFPSO, RBFNN, and MLANN are coded in MATLAB as per the design and the

learning algorithm described earlier. Simulation studies are carried out with a different input features combination to test the sensitivity of the soft computing approach to control parameters until a satisfactory accuracy level is achieved for estimation of $FAO56-ET_0$ for different study locations. Detailed information regarding modeling strategies and respective control parameters that produce optimum results during the simulation process are shown in Table 4 for different soft computing models.

Calibration of RBFDE, RBFPSO, RBFNN, and MLANN models is done using the above-listed network parameters with training datasets of all the three study locations, Raipur, Jagdalpur, and Ambikapur. During the training process, input patterns are given to the model sequentially and the corresponding estimated output is obtained at the output layer after completion of the forward pass (Fig. 3). The estimated output is compared with the corresponding target $FAO56-ET_0$ output to compute the instantaneous error cost function. Real-time update of the model parameters is done in each instance to minimize the squared error using respective evolutionary (DE and PSO) and conventional back-propagation learning algorithms (RBFNN and MLANN). The process continues until all the available training input patterns for model calibration gets

Table 3 Input feature combinations used in soft computing and empirical models

Type	Soft computing models				Equivalent empirical model	Input feature combinations
	RBFDE	RBFPSO	RBFNN	MLANN		
I	RBFDE1	RBFPSO1	RBFNN1	MLANN1	Hargreaves	T_{max} , T_{min}
II	RBFDE2	RBFPSO2	RBFNN2	MLANN2	Turc	T_{max} , T_{min} , BSS
III	RBFDE3	RBFPSO3	RBFNN3	MLANN3	Open Pan	EP, RH_1 & RH_2 , WS
IV	RBFDE4	RBFPSO4	RBFNN4	MLANN4	Blaney-Criddle	T_{max} , T_{min} , RH_1 & RH_2 , WS, BSS
V	RBFDE5	RBFPSO5	RBFNN5	MLANN5	–	T_{max} , T_{min} , BSS, WS

Table 4 Parameters of the soft computing models used for simulation

Model parameters	RBFDE	RBFPSO	RBFNN	MLANN
Dimension of input features (D)	2, 3, 4 & 6	2,3,4 & 6	2, 3 4 & 6	2,3,4 & 6
Normalization of input features	0 to 1	0 to 1	0 to 1	– 1 to 1
Normalization method	$\frac{x_k - x_{min}}{x_{min} - x_{max}}$, where x_k represents k th instance			
Initial values of the model parameters	0 to 1	0 to 1	0 to 1	– 1 to 1
Number of centers for RBF based models (c)	10	10	10	–
Number of neurons in intermediate layer	–	–	–	5
Output neuron	1	1	1	1
Activation function	Gaussian	Gaussian	Gaussian	\tanh
Learning algorithm	DE	PSO	RBF update rules	Back-propagation
Convergence coefficients/control parameters/	F=0.9 CR=0.9	Constriction coefficient $C_1 = C_2 = 1.49$ Inertia weight w Linearly decreased from 0.9 to 0.4	$\eta_1 = \eta_2 = \eta_3 = 0.01$	$\eta = 0.01$
Number of population/particles	$5 \times d$	50	–	–

exhausted. This completes one cycle called an epoch. At the end of each epoch, the mean square error is computed and stored for each epoch to examine the learning characteristic of soft computing models. The iterative process is repeated several times until MSE is minimized to a desired low value nearly close to zero. This completes the supervised learning process and model parameters are then fixed to constitute soft computing models. A similar calibration process is adopted for all soft computing approaches.

To test the performance of different soft computing models, test data patterns are then presented sequentially at the input layer of the model and through forward pass respective estimated ET_0 is obtained at the output layer for all the test patterns. These ET_0 estimates are then compared with corresponding target FAO56-PM ET_0 values. Performance evaluation measures, MAPE (%), RMSE ($mm\ week^{-1}$), R^2 and NSE as described in the previous section are then computed using desired and estimated output of different types of soft computing models and equivalent empirical approaches for comparison of model performance, which ultimately leads to model selection. The computed values of performance evaluation measures for different types of soft computing models and

equivalent empirical approaches considered are listed in Tables 5 and 6 for all three locations. Comparative results of the analysis are discussed below:

- i. For type I soft computing models, MAPE ranges from lowest of 7.4 for RBFDE1 and RBFDE2 (at Raipur) to highest of 11.8 for MLANN1 (at Jagdalpur), whereas MAPE obtained with Hargreaves model is comparatively very high and ranges between 22.6 (at Raipur) to 30.3 (at Ambikapur).
- ii. Type II soft computing models produce improved ET_0 estimates with low MAPE compared to type I models. For type II models, MAPE ranges from the lowest of 4.9 with RBFDE2 (at Raipur) to a high of 10.2 with MLANN2 (at Raipur). MAPE is again quite higher with the equivalent empirical approach of Turc, which is obtained between 10.1 (at Jagdalpur) to 13.9 (at Raipur).
- iii. Subsequently, for type III models, MAPE values are computed close to that of type II models, which varied between a lowest of 4.7 with RBFDE3 & RBFPSO3 (at Raipur) to a high of 8.0 with MLANN3 (at Jagdalpur). MAPE for the Open Pan approach varies between 12.2 (at Raipur) to 22.2 (at Ambikapur),

Table 5 MAPE (%) and RMSE ($mm\ week^{-1}$) for different types of soft computing and equivalent empirical models with test data sets of Ambikapur, Jagdalpur, and Raipur

Type	MODEL	Ambikapur		Jagdalpur		Raipur	
		MAPE	RMSE	MAPE	RMSE	MAPE	RMSE
I	RBFDE1	8.9*	2.96	10.6	3.63	7.4	2.98
	RBFPSO1	9.3	3.04	10.7	3.67	7.4	2.98
	RBFNN1	9.3	3.06	10.5	3.79	8.5	3.22
	MLANN1	10.0	3.11	11.8	3.88	9.0	3.23
	Hargreaves	30.3	7.15	29.0	7.53	22.6	6.13
II	RBFDE2	4.9	2.10	5.3	2.74	5.7	2.63
	RBFPSO2	5.0	2.10	5.4	2.78	5.9	2.68
	RBFNN2	4.9	2.23	5.2	2.81	5.9	3.14
	MLANN2	5.1	2.31	6.2	2.93	10.2	3.45
	Turc	13.9	3.73	10.1	4.26	13.9	6.54
III	RBFDE3	5.9	1.82	7.0	2.18	4.7	1.75
	RBFPSO3	6.0	1.89	6.9	2.21	4.7	1.77
	RBFNN3	6.3	1.95	7.0	2.27	4.9	1.80
	MLANN3	7.8	2.43	8.0	2.48	6.6	2.28
	Open-Pan	22.2	6.55	21.7	6.30	16.2	5.05
IV	RBFDE4	2.1	0.68	3.7	1.06	1.1	0.36
	RBFPSO4	3.4	1.02	3.8	1.10	1.3	0.43
	RBFNN4	4.4	1.32	4.1	1.22	2.4	0.82
	MLANN4	4.6	1.36	4.4	1.25	3.9	1.29
	Blaney-Criddle	22.1	6.12	15.5	4.27	22.6	7.20
V	RBFDE5	2.2	0.66	3.4	1.06	1.9	0.80
	RBFPSO5	3.9	1.08	3.6	1.08	2.2	0.87
	RBFNN5	4.6	1.38	4.7	1.39	2.9	1.07
	MLANN5	5.2	1.48	5.2	1.46	4.1	1.31

*boldface numbers highlight the best results

Table 6 R² and NSE for different types of soft computing and equivalent empirical models with test datasets of Ambikapur, Jagdalpur, and Raipur

Type	MODEL	Ambikapur		Jagdalpur		Raipur	
		R ²	NSE	R ²	NSE	R ²	NSE
I	RBFDE1	0.949	0.892*	0.878	0.785	0.955	0.932
	RBFPSO1	0.950	0.886	0.880	0.781	0.954	0.933
	RBFNN1	0.950	0.884	0.876	0.766	0.948	0.922
	MLANN1	0.938	0.881	0.871	0.755	0.946	0.921
	Hargreaves	0.933	0.371	0.809	0.077	0.900	0.716
II	RBFDE2	0.981	0.946	0.955	0.950	0.969	0.948
	RBFPSO2	0.980	0.945	0.954	0.948	0.969	0.946
	RBFNN2	0.978	0.939	0.950	0.947	0.946	0.926
	MLANN2	0.976	0.934	0.948	0.939	0.929	0.910
	Turc	0.905	0.829	0.795	0.704	0.833	0.677
III	RBFDE3	0.960	0.959	0.934	0.922	0.978	0.977
	RBFPSO3	0.956	0.956	0.930	0.920	0.977	0.976
	RBFNN3	0.954	0.953	0.931	0.916	0.977	0.976
	MLANN3	0.928	0.927	0.908	0.900	0.961	0.961
	Open-Pan	0.743	0.472	0.827	0.353	0.947	0.808
IV	RBFDE4	0.995	0.994	0.988	0.982	0.999	0.999
	RBFPSO4	0.989	0.987	0.986	0.980	0.999	0.999
	RBFNN4	0.982	0.978	0.985	0.976	0.996	0.995
	MLANN4	0.982	0.977	0.983	0.975	0.990	0.988
	Blaney-Criddle	0.826	0.538	0.827	0.703	0.821	0.608
V	RBFDE5	0.995	0.995	0.983	0.982	0.996	0.995
	RBFPSO5	0.986	0.986	0.983	0.981	0.995	0.994
	RBFNN5	0.980	0.976	0.973	0.969	0.993	0.991
	MLANN5	0.974	0.973	0.974	0.965	0.988	0.987

*boldface numbers highlight the best results

which is very high as compared to type III soft computing models.

- iv. Type IV soft computing models yield better results as compared to all other types of soft computing and empirical models. MAPE ranges between a low of 1.1 to a high of 3.9 at Raipur, followed by 3.7 to 4.4 at Jagdalpur and 2.2 to 4.6 at Ambikapur with RBFDE4 and MLANN4 respectively. MAPE with the Blaney-Criddle method is again quite inferior as compared to type IV soft computing approaches and ranges from 15.5 (at Jagdalpur) to 22.6 (at Raipur).
- v. Type V models also produced good results, as reasonably fair estimates of ET₀ can be obtained between a low MAPE of 1.9 with RBFDE5 (at Raipur) to 5.2 with MLANN5 (at Jagdalpur and Ambikapur), which is very much comparable to that of type IV models, even without taking humidity data as one of the input features.
- vi. Regarding RMSE, type I soft computing models have resulted in RMSE between 2.98 mm week⁻¹ (at Raipur) with RBFDE1 and RBFPSO1 to 3.88 week⁻¹ (at Jagdalpur) with MLANN1, as against the higher RMSE of 6.13 week⁻¹ (at Raipur) to 7.53 mm week⁻¹ (at Jagdalpur) obtained with Hargreaves approaches.
- vii. Type II soft computing models have produced improved RMSE as compared to type I models, which ranges between a low of 2.10 mm week⁻¹ with RBFDE2 (at Ambikapur) to a high of 3.45 mm week⁻¹ with MLANN2 (at Raipur). Interestingly, at Jagdalpur the soft computing models produce comparatively better estimates of FAO56-PM ET₀ in terms of RMSE as compared with similar models at Jagdalpur and Raipur. In general, type II soft computing models have yielded better ET₀ estimates as compared to Turc methods, for which RMSE ranges between 3.73 (at Ambikapur) to 6.54 mm week⁻¹ (at Raipur).
- viii. Regarding type III models, RMSE has improved further and is computed between a low of 1.75 mm week⁻¹ with RBFDE3 to a high of 2.28 mm week⁻¹ with MLANN3 at Raipur, whereas the same for Jagdalpur and Ambikapur, it varied between a low of 1.82 mm week⁻¹ with RBFDE3 to a high of 2.48 mm week⁻¹ with MLANN3. The equivalent empirical method of Open Pan has produced higher RMSE, which varied between 5.05 mm week⁻¹

in Raipur to 6.55 mm week⁻¹ at Ambikapur, which is almost three times more as compared to type III soft computing models.

- ix. Similar to MAPE, type IV soft computing models have yielded excellent results in terms of RMSE also. In Raipur, RMSE ranges between the lowest of 0.36 mm week⁻¹ with RBFDE4 to the highest of 1.29 mm week⁻¹ with MLANN4. At Jagdalpur, it ranges between 1.06 mm week⁻¹ with RBFDE4 to 1.25 mm week⁻¹ with MLANN4, whereas at Ambikapur, RMSE ranges between 0.68 mm week⁻¹ with RBFDE4 to 1.36 mm week⁻¹ with MLANN4. The low RMSE values (< 1 mm week⁻¹) obtained with evo-

lutionary optimized hybrid soft computing models (RBFDE4 and RBFDE5) are quite encouraging. This demonstrates the potential of the RBFDE4 and RBFPSO4 models and these models may consider as an alternative to the FAO56-PM empirical approach for ET₀ estimation in the study area. In contrast, Blaney-Cridde has produced very high RMSE, which ranges between 4.27 to 7.29 mm/week at different locations, similar to that of the Open Pan method.

- x. Type V, soft computing models have also produced better results which is quite identical with type IV models even without including humidity data as an input feature. RMSE with type V models ranges

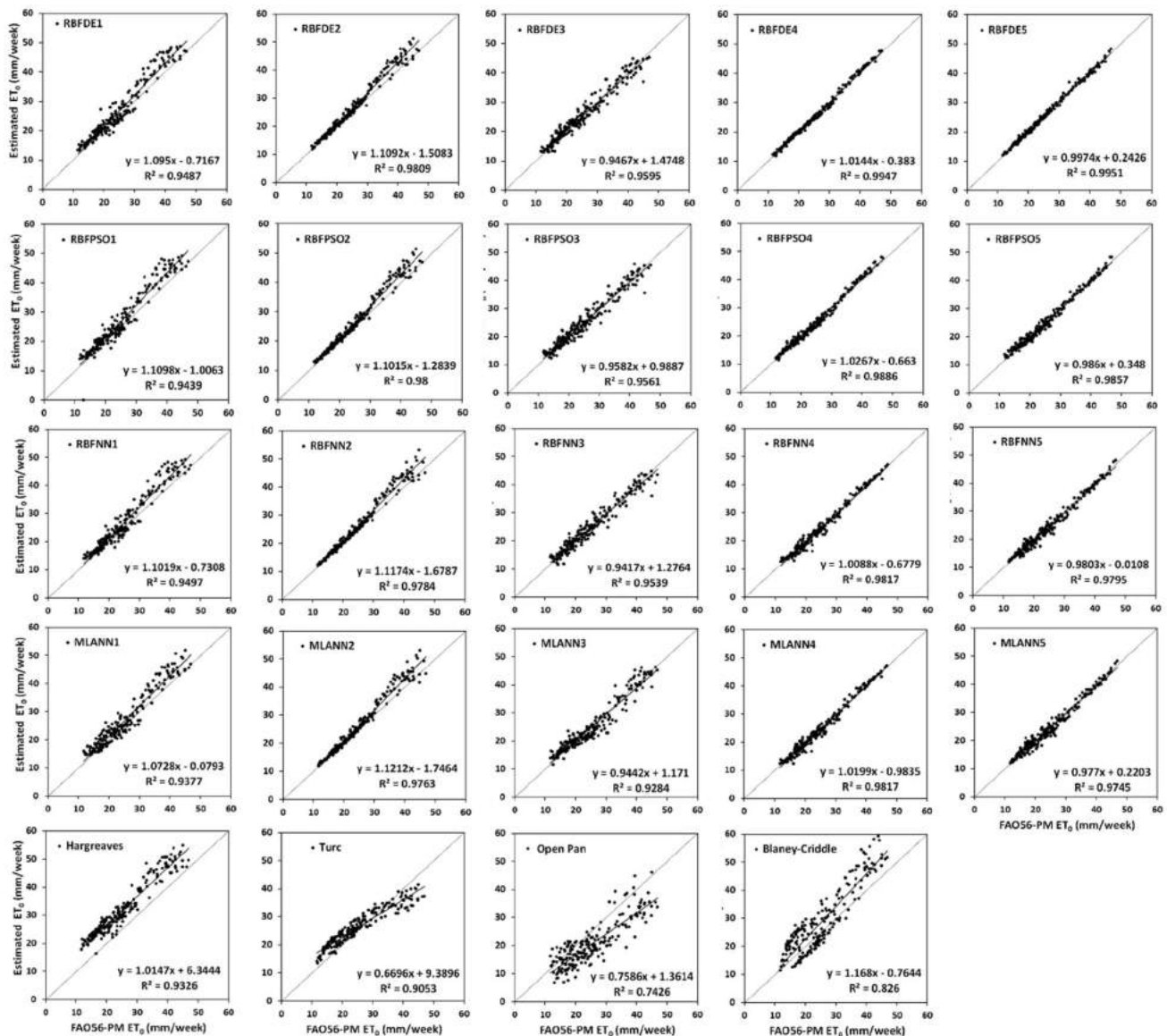


Fig. 6 Relationship between estimated ET₀ and FAO56-PM ET₀ for different soft computing and empirical models with test data sets at Ambikapur

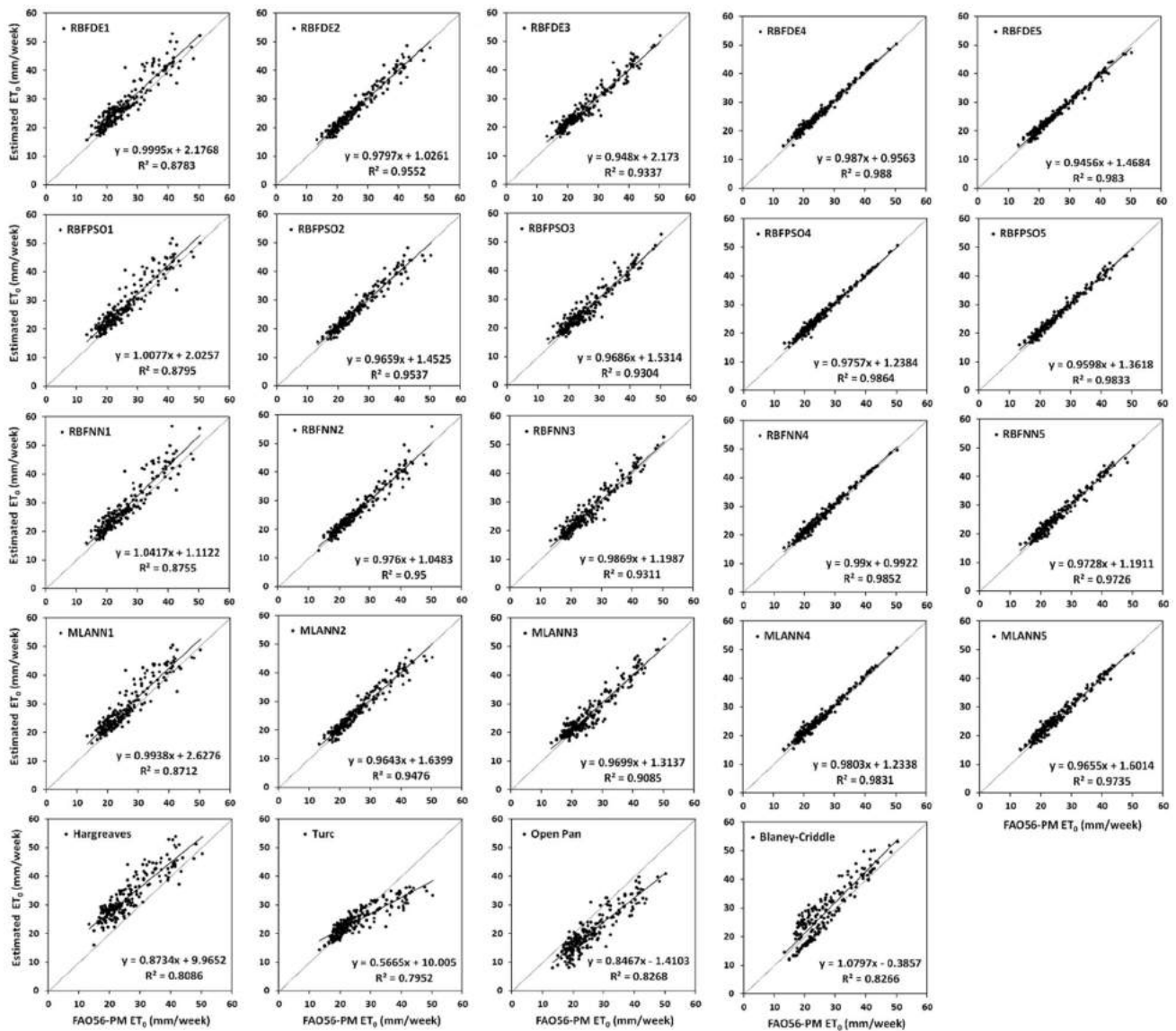


Fig. 7 Relationship between estimated ET_0 and FAO56-PM ET_0 for different soft computing and empirical models with test data sets at Jagdalpur

between lowest $0.66 \text{ mm week}^{-1}$ with RBFDE5 to highest of $1.48 \text{ mm week}^{-1}$ with MLANN5 at Ambikapur. At Raipur and Jagdalpur, RMSE ranges between 0.80 to 1.31 and 1.06 to $1.46 \text{ mm week}^{-1}$ with RBFDE5 and MLANN5 models respectively.

To further examine the relationship between the estimated and FAO56-PM ET_0 , two more statistical measures, R^2 and NSE are computed for different soft computing and empirical models and shown in Table 6. The linear relationship between estimated ET_0 and FAO56- ET_0 is also depicted in Figs. 6, 7, and 8 in Ambikapur, Jagdalpur, and Raipur respectively. In general, both R^2 and NSE convey similar information about the model performance and therefore,

the marginal difference is observed between these two performance evaluation measures within a similar type of model in different locations. However, sometimes R^2 values give a false indication and produce higher values close to 1 despite a very high intercept. In such cases corresponding NSE helps in evaluating the model performance.

- xi. Type IV soft computing models have produced better R^2 and NSE values as compared to all other models considered for investigation. The highest R^2 values of 0.999, 0.988, and 0.995 are obtained with RBFDE4 and RBFPSO4 in Raipur, Jagdalpur, and Ambikapur respectively with test data sets. The remaining type IV soft computing models also pro-

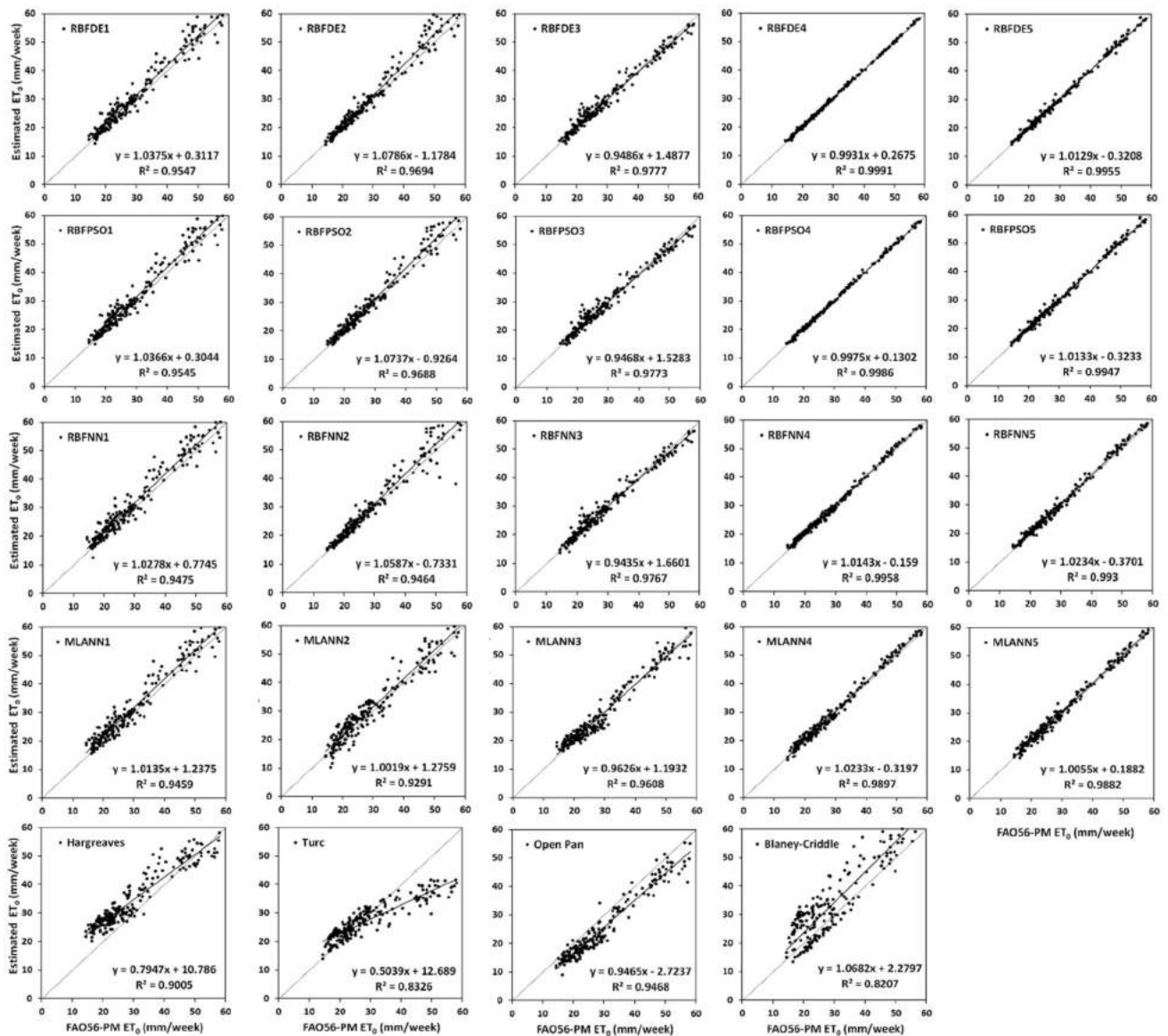


Fig. 8 Relationship between estimated ET_0 and FAO56-PM ET_0 for different soft computing and empirical models with test data sets at Raipur

duced good R^2 and NSE values as compared to the equivalent empirical approach.

- xii. Similar results are obtained with type V soft computing models, as R^2 and the corresponding NSE vary between 0.965 to 0.996 in different locations, with RBFDE5 and RBFPSO5 being the best models.
- xiii. For the type III model, R^2 and NSE range between 0.961 to 0.978 at Raipur and between 0.900 to 0.960 at Jagdalpur and Ambikapur. R^2 and NSE with type II soft computing models vary between 0.910 to 0.981. Hence, it can be stated that consistent ET_0 estimates with a fair degree of agreement between estimated

and target ET_0 can be obtained using Type both II and III soft computing models.

- xiv. Type I soft computing models of RBFDE1 and RBFPSO1 have resulted in slightly lower R^2 and NSE values than RBFNN1 and MLANN1 as compared to remaining types mainly as fewer input features are involved in computations.
- xv. Inconsistence and low R^2 and NSE values that have obtained with empirical approaches of Hargreaves, Turc, Open Pan, and Blaney Criddle as compared to their equivalent soft computing models of respective types, clearly establish the fact that soft comput-

ing models produce better estimates of FAO56-PM ET_0 than empirical models.

Results of the performance evaluation analysis indicate that the evolutionary optimized hybrid soft computing models considered for the investigation (RBFDE and RBFPSO) performed consistently better than other conventional soft computing techniques (RBFNN and MLANN) and empirical approaches in all the objectives. From the inferences, it is also evident that when a complete set of the climatic variable is involved in the computation of ET_0 using these models, it looks very difficult to choose between RBFDE and RBFPSO as they look statistically similar in some cases. However, the proposed RBFDE is recommended because of its preciseness and generalization performance in estimating ET_0 in all the stations considered for the study.

4 Conclusions ET_0

The present investigation is carried out to examine the generalized potential of evolutionary optimized hybrid soft computing techniques RBFDE and RBFPSO for the estimation of ET_0 in different ACZs. The ET_0 estimates obtained with proposed RBFDE and RBFPSO models are compared to the conventional neural network (RBFNN, MLANN) and existing empirical approaches. Looking to the scarcity of complete datasets required for computation of FAO-PM ET_0 , four variants of each category of soft computing models (RBFDE, RBFPSO, RBFNN, and MLANN) equivalent (in terms of input feature combination) to empirical approaches (Hargreaves, Turc, Open Pan and Blaney-Criddle) is examined. It can be concluded that different soft computing models considered in this investigation, have resulted in improved and more consistent FAO56-PM ET_0 estimates as compared to equivalent empirical approaches. Among the soft computing models, evolutionary models RBFDE and RBFPSO produced a more precise estimation of FAO56-PM ET_0 than conventional RBFNN and MLANN as proposed RBFDE and RBFPSO models resulted in low MAPE and RMSE and high R^2 and NSE close to 1 in most of the cases. However, ET_0 estimates obtained with the proposed RBFDE seems to be slightly better than RBFPSO. Hence, appropriate soft computing models may be recommended for the estimation of ET_0 in other stations of respective ACZs of the study area. The proposed soft computing models may be embedded in crop weather simulation models as subroutines for precise estimation ET_0 with available input features. However, re-calibration and re-validation of these data-driven models are essentially required for their effective implantation in other parts of the world.

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest.

Ethical approval This article does not contain any studies with human participants or animals performed by any of the authors.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

1. Allen RG, Pereira LS, Raes D, Smith M (1998) Introduction to evapotranspiration. In: Crop evapotranspiration-guidelines for computing crop water requirements-FAO Irrigation and drainage paper, vol 56, pp 1–13. FAO, Rome. <https://doi.org/10.1016/j.eja.2010.12.001>
2. Elizabeth AH, Robert EC (2013) Water balance estimates of evapotranspiration rates in areas with varying land use. In: Evapotranspiration-an overview. InTech. <https://doi.org/10.5772/52811>
3. Allen RG, Jensen ME, Wright JL, Burman RD (1989) Operational estimates of reference evapotranspiration. *Agron J* 81(4):650–662. <https://doi.org/10.2134/agronj1989.00021962008100040019x>
4. Droogers P, Allen RG (2002) Estimating reference evapotranspiration under inaccurate data conditions. *Irrigat Drain Syst* 16(1):33–45. <https://doi.org/10.1023/A:1015508322413>
5. Hargreaves GL, Hargreaves GH, Riley JP (1985) Agricultural, benefits for Senegal River Basin. *J Irrig Drain Eng ASCE* 111:113–124
6. Turc L (1961) Evaluation des besoins en eau d'irrigation, evapotranspiration potentielle, formule climatique simplifiée et mise a jour. *Ann Agron* 12:13–49 (in French)
7. Doorenbos J, Pruitt WO (1977) Guidelines for predicting crop water requirements. Irrigation and drainage paper No 24, 2nd edn, Food and Agriculture Organization, Rome, p 156
8. Christiansen JE (1968) Pan evaporation and evapotranspiration by climatic data. *J Irrig Drain Div Am Soc Civil Eng* 94:243–263
9. Bapuji RB, Sandeep VM, Rao VUM, Venkateswarlu B (2012) Potential Evapotranspiration estimation for Indian conditions: Improving accuracy through calibration coefficients. Technical Bull No 1/2012. All India Co-ordinated Research Project on Agrometeorology, Central Research Institute for Dryland Agriculture, Hyderabad, p 60
10. Chauhan S, Shrivastava RK (2009) Performance evaluation of reference evapotranspiration estimation using climate based methods and artificial neural networks. *Water Resour Manag* 23(5):825–837

11. Kisi O, Guven A (2010) Evapotranspiration modeling using linear genetic programming technique. *J Irrig Drain Eng* 136(10):715–723. [https://doi.org/10.1061/\(asce\)ir.1943-4774.0000244](https://doi.org/10.1061/(asce)ir.1943-4774.0000244)
12. Kumar M, Raghuwanshi NS, Singh R (2011) Artificial neural networks approach in evapotranspiration modeling: a review. *Irrig Sci*. <https://doi.org/10.1007/s00271-010-0230-8>
13. Mallikarjuna P, Jyothy SA, Sekhar Reddy KC (2012) Daily reference evapotranspiration estimation using linear regression and ANN models. *J Inst Eng Ser A* 93(4):215–221. <https://doi.org/10.1007/s40030-013-0030-2>
14. Kisi O (2016) Modeling reference evapotranspiration using three different heuristic regression approaches. *Agric Water Manag* 169:162–172. <https://doi.org/10.1016/j.agwat.2016.02.026>
15. Yassin MA, Alazba AA, Mattar MA (2016) Artificial neural networks versus gene expression programming for estimating reference evapotranspiration in an arid climate. *Agric Water Manag* 163:110–124. <https://doi.org/10.1016/j.agwat.2015.09.009>
16. Antonopoulos VZ, Antonopoulos AV (2017) Daily reference evapotranspiration estimates by artificial neural networks technique and empirical equations using limited input climate variables. *Comput Electron Agric* 132:86–96. <https://doi.org/10.1016/j.compag.2016.11.011>
17. Nema MK, Khare D, Chandniha SK (2017) Application of artificial intelligence to estimate the reference evapotranspiration in sub-humid Doon valley. *Appl Water Sci* 7(7):3903–3910. <https://doi.org/10.1007/s13201-017-0543-3>
18. Pandey PK, Nyori T, Pandey V (2017) Estimation of reference evapotranspiration using data-driven techniques under limited data conditions. *Model Earth Syst Environ* 3(4):1449–1461. <https://doi.org/10.1007/s40808-017-0367-z>
19. Patil AP, Deka PC (2017) Performance evaluation of hybrid Wavelet-ANN and Wavelet-ANFIS models for estimating evapotranspiration in arid regions of India. *Neural Comput Appl* 28(2):275–285. <https://doi.org/10.1007/s00521-015-2055-0>
20. Wen X, Si J, He Z, Wu J, Shao H, Yu H (2015) Support-vector-machine-based models for modeling daily reference evapotranspiration with limited climatic data in extreme arid regions. *Water Resour Manag* 29(9):3195–3209. <https://doi.org/10.1007/s11269-015-0990-2>
21. Partal T (2016) Comparison of wavelet-based hybrid models for daily evapotranspiration estimation using meteorological data. *KSCE J Civil Eng* 20(5):2050–2058. <https://doi.org/10.1007/s12205-015-0556-0>
22. Kisi O, Demir V (2016) Evapotranspiration estimation using six different multi-layer perceptron algorithms. *Irrig Drain Syst Eng*. <https://doi.org/10.4172/2168-9768.1000164>
23. Dou X, Yang Y (2018) Evapotranspiration estimation using four different machine learning approaches in different terrestrial ecosystems. *Comput Electron Agric* 148:95–106. <https://doi.org/10.1016/j.compag.2018.03.010>
24. Adamala S (2018) Temperature based generalized wavelet-neural network models to estimate evapotranspiration in India. *Inf Process Agric* 5(1):149–155. <https://doi.org/10.1016/j.inpa.2017.09.004>
25. Sanikhani H, Kisi O, Maroufpoor E, Yaseen ZM (2018) Temperature-based modeling of reference evapotranspiration using several artificial intelligence models: application of different modeling scenarios. *Theoret Appl Climatol* 135(1–2):449–462. <https://doi.org/10.1007/s00704-018-2390-z>
26. Ozkan C, Kisi O, Akay B (2011) Neural networks with artificial bee colony algorithm for modeling daily reference evapotranspiration. *Irrig Sci* 29(6):431–441. <https://doi.org/10.1007/s00271-010-0254-0>
27. Eslamian SS, Gohari SA, Zareian MJ, Firoozfar A (2012) Estimating Penman-Monteith reference evapotranspiration using artificial neural networks and genetic algorithm: a case study. *Arab J Sci Eng* 37(4):935–944. <https://doi.org/10.1007/s13369-012-0214-5>
28. Aghajanoloo MB, Sabziparvar AA, Hosseinzadeh TP (2013) Artificial neural network-genetic algorithm for estimation of crop evapotranspiration in a semi-arid region of Iran. *Neural Comput Appl* 23(5):1387–1393. <https://doi.org/10.1007/s00521-012-1087-y>
29. Petković D, Gocić M, Shamshirband S, Qasem SN, Trajković S (2016) Particle swarm optimization-based radial basis function network for estimation of reference evapotranspiration. *Theoret Appl Climatol* 125(3–4):555–563. <https://doi.org/10.1007/s00704-015-1522-y>
30. Jovic S, Nedeljkovic B, Golubovic Z, Kostic N (2018) Evolutionary algorithm for reference evapotranspiration analysis. *Comput Electron Agric* 150:1–4. <https://doi.org/10.1016/j.compag.2018.04.003>
31. Feng Y, Cui N, Zhao L, Hu X, Gong D (2016) Comparison of ELM, GANN, WNN and empirical models for estimating reference evapotranspiration in the humid region of Southwest China. *J Hydrol* 536:376–383. <https://doi.org/10.1016/j.jhydrol.2016.02.053>
32. Gocić M, Motamedi S, Shamshirband S, Petković D, Ch S, Hashim R, Arif M (2015) Soft computing approaches for forecasting reference evapotranspiration. *Comput Electron Agric* 113:164–173. <https://doi.org/10.1016/j.compag.2015.02.010>
33. Mehdizadeh S, Behmanesh J, Khalili K (2017) Using MARS, SVM, GEP and empirical equations for estimation of monthly mean reference evapotranspiration. *Comput Electron Agric* 139:103–114. <https://doi.org/10.1016/j.compag.2017.05.002>
34. Mattar MA, Alazba AA (2018) GEP and MLR approach for the prediction of reference evapotranspiration. *Neural Comput Appl*. <https://doi.org/10.1007/s00521-018-3410-8>
35. Baltzis K (2013) Patented applications of differential evolution in microwave and communication engineering. *Recent Patents Comput Sci* 6(2):115–128. <https://doi.org/10.2174/22132759113069990004>
36. Tenaglia GC, Lebensztajn L (2014) A multiobjective approach of differential evolution optimization applied to electromagnetic problems. *IEEE Trans Magn*. <https://doi.org/10.1109/TMAG.2013.2285980>
37. Uher V, Gajdoš P, Radecký M, Snášel V (2016) Utilization of the discrete differential evolution for optimization in multidimensional point clouds. *Comput Intell Neurosci*. <https://doi.org/10.1155/2016/6329530>
38. Das S, Abraham A, Konar A (2008) Particle swarm optimization and differential evolution algorithms: technical analysis, applications and hybridization perspectives. *Stud Comput Intell* 116:1–38. https://doi.org/10.1007/978-3-540-78297-1_1
39. Hui S, Suganthan PN (2016) Ensemble and arithmetic combination-based speciation differential evolution for multimodal optimization. *IEEE Trans Cybern* 46(1):64–74. <https://doi.org/10.1109/TCYB.2015.2394466>
40. Ramadas M, Abraham A, Kumar S (2018) RDE reconstructed mutation strategy for differential evolution algorithm. *Adv Intell Syst Comput* 614:76–85. https://doi.org/10.1007/978-3-319-60618-7_8
41. Ramadas M, Pant M, Abraham A, Kumar S (2019) Segmentation of weather radar image based on hazard severity using RDE: reconstructed mutation strategy for differential evolution algorithm. *Neural Comput Appl* 31:1253–1261. <https://doi.org/10.1007/s00521-017-3091-8>
42. Ghosh A, Mallipeddi R, Das S, Das AK (2018) A switched parameter differential evolution with multi-donor mutation and annealing based local search for optimization of lennard-jones atomic clusters. In: 2018 IEEE congress on evolutionary computation,

- CEC 2018-Proceedings. Institute of Electrical and Electronics Engineers Inc. <https://doi.org/10.1109/CEC.2018.8477991>
43. Biswas PP, Suganthan PN, Mallipeddi R, Amaratunga GAJ (2018) Optimal power flow solutions using differential evolution algorithm integrated with effective constraint handling techniques. *Eng Appl Artif Intell* 68:81–100. <https://doi.org/10.1016/j.engappai.2017.10.019>
 44. Rout M, Majhi B, Majhi R, Panda G (2014) Forecasting of currency exchange rates using an adaptive ARMA model with differential evolution based training. *J King Saud Univ Comput Inf Sci* 26(1):7–18. <https://doi.org/10.1016/j.jksuci.2013.01.002>
 45. Pujari P, Majhi B (2017) Application of nature-inspired technique to Odia handwritten numeral recognition, book chapter in handbook of research on modeling, analysis and application of nature-inspired Metaheuristic algorithms, IGI Global Publication, USA, pp 377–399 <https://doi.org/10.4018/978-1-5225-2857-9.ch019>
 46. World Meteorological Organization (2012) WMO-No. 8-Guide to meteorological instruments and methods of observation, pp I.8-1–I.9-1
 47. Broomhead D, Lowe D (1988) Multivariable functional interpolation and adaptive networks. *Complex Syst* 2:321–355
 48. Fernández-Redondo M, Hernández-Espinosa C, Ortiz-Gómez M, Torres-Sospedra J (2004) Training radial basis functions by gradient descent. In: *Lecture notes in artificial intelligence (subseries of lecture notes in computer science)*, Vol 3070, pp 184–189. Springer Verlag. https://doi.org/10.1007/978-3-540-24844-6_23
 49. Storn R, Price K (1995) Differential evolution—a simple and efficient adaptive scheme for global optimization over continuous spaces. Technical Report TR-95-012, pp 1–12 <https://doi.org/10.1.1.1.9696>
 50. Storn R, Price K (1997) Differential evolution—a simple and efficient Heuristic for global optimization over continuous spaces. *J Global Optim* 11(4):341–359. <https://doi.org/10.1023/A:1008202821328>
 51. Kennedy J, Eberhart R (1995) Particle swarm optimization. *Proc IEEE Int Conf Neural Netw* 4:1942–1948
 52. Eberhart R, Kennedy J (2002) A new optimizer using particle swarm theory. In: *Institute of Electrical and Electronics Engineers (IEEE)*, pp 39–43 <https://doi.org/10.1109/mhs.1995.494215>
 53. Eberhart R, Yuhui S (2002) Particle swarm optimization: developments, applications and resources. In: *Institute of Electrical and Electronics Engineers (IEEE)*, pp 81–86 <https://doi.org/10.1109/cec.2001.934374>
 54. del Valle Y, Venayagamoorthy GK, Mohagheghi S, Hernandez JC, Harley RG (2008) Particle swarm optimization: basic concepts, variants and applications in power systems. *IEEE Trans Evol Comput*. <https://doi.org/10.1109/TEVC.2007.896686>
 55. Chen LF, Su CT, Chen KH (2012) An improved particle swarm optimization for feature selection. *Intell Data Anal* 16(2):167–182. <https://doi.org/10.3233/IDA-2012-0517>
 56. Zhang Y, Wang S, Ji GA (2015) Comprehensive survey on particle Swarm optimization algorithm and its applications. *Mathematical Problems in Engineering*, pp 1–38 <https://doi.org/10.1155/2015/931256>
 57. Haykin S (1998) *Neural networks—a comprehensive foundation*, 2nd edn. Prentice-Hall, Upper Saddle River, pp 26–32
 58. Nash JE, Sutcliffe JV (1970) River flow forecasting through conceptual models part I—a discussion of principles. *J Hydrol* 10(3):282–290
- Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Modelling economic policy issues

Impact of COVID-19 on GDP of major economies: Application of the artificial neural network forecaster

Pradyot Ranjan Jena ^{a,*}, Ritanjali Majhi ^a, Rajesh Kalli ^b, Shunsuke Managi ^c, Babita Majhi ^d

^a School of Management, National Institute of Technology Karnataka, Surathkal, Mangalore 575025, India

^b Department of Management Studies, Madanapalle Institute of Technology & Science, India

^c Urban Institute & Department of Civil Engineering, Kyushu University, Japan

^d Department of CSIT, Guru Ghasidas Vishwavidyalaya, Central University, Bilaspur, India



ARTICLE INFO

Article history:

Received 8 August 2020

Received in revised form 9 December 2020

Accepted 10 December 2020

Available online 18 December 2020

Keywords:

Forecasting

GDP

Multilayer neural network

Mean absolute percentage error

ABSTRACT

The ongoing COVID-19 pandemic has caused global health impacts, and governments have restricted movements to a certain extent. Such restrictions have led to disruptions in economic activities. In this paper, the GDP figures for the April–June quarter of 2020 for eight countries, namely, the United States, Mexico, Germany, Italy, Spain, France, India, and Japan, are forecasted. Considering that artificial neural network models have higher forecasting accuracy than statistical methods, a multilayer artificial neural network model is developed in this paper. This model splits the dataset into two parts: the first with 80% of the observations and the second with 20%. The model then uses the first part to optimize the forecasting accuracy and then applies the optimized parameters to the second part of the dataset to assess the model performance. A forecasting error of less than 2% is achieved by the model during the testing procedure. The forecasted GDP figures show that the April–June quarter of the current year experienced sharp declines in GDP for all countries. Moreover, the annualized GDP growth is expected to reach double-digit negative growth rates. Such alarming prospects require urgent rescue actions by governments.

© 2020 Economic Society of Australia, Queensland. Published by Elsevier B.V. All rights reserved.

1. Introduction

The novel coronavirus disease 2019 (COVID-19), which first appeared in Wuhan city, China, in December 2019, has caused global distress, claiming lives and collapsing economies, as many individuals are connected globally (Acemoglu et al., 2020; Nakamura and Managi, 2020). Given its deepening threat to human lives and economies, the Director-General of the World Health Organization (WHO) declared COVID-19 a Public Health Emergency of International Concern (WHO, 2020) on 30th January 2020. Policymakers in every country are under pressure to maintain a balance between containing the disease by implementing lockdowns and saving the jobs and livelihoods of a large number of people by keeping economic activities undeterred (Yoo and Managi, 2020). Restrictions on the movements of both people and goods have disrupted supply chains and accelerated the unemployment problem. Given this consideration, it has become important for countries to assess the broad economic implications of COVID-19. Macroeconomic indicators represent the health and stability of a country's economy. Gross domestic product (GDP), the most widely recognized indicator, accounts for the

* Corresponding author.

E-mail address: pradyotjena@nitk.edu.in (P.R. Jena).

overall goods and services produced within a country. Accurate GDP projections can equip policymakers with tools to effectively plan for future economic development. In this paper, we develop a multilayer neural network model that can forecast GDP with minimum error.

Previous methodologies employed in GDP forecasting can be classified into two broad categories. The first emphasizes a strong theoretical background and applies linear models (Guégan and Rakotomaroahy, 2010). However, concerns with this approach have been raised due to strong hypotheses on model specification, estimation, and asymptotic properties of the estimated parameters (Guégan and Rakotomaroahy, 2010). The second, nonlinear models, includes the nearest-neighbors method and neural network (Härdle et al., 2012; Tkacz, 2001a; Kock and Teräsvirta, 2014). The nearest-neighbors method uses parametric models and neural networks for nonparametric modeling. Forecasting GDP using a linear autoregressive model based on the Box–Jenkins approach or multivariate VAR (Box et al., 2015; Bařbura et al., 2010; Balçilar et al., 2015) is common. Additionally, artificial neural networks (ANNs) have been applied to forecast macroeconomic indicators such as inflation, exchange rates, oil prices, and interest rates (Hlaváček et al., 2005; Jena et al., 2015; Ali Choudhary and Haider, 2012; McNelis, 2005; Ramos-Pérez et al., 2019; Szafranek, 2019). Furthermore, ANN models have been used to forecast GDP in different countries, such as the United States (Loermann and Maas, 2019), China (Shi et al., 2006), Sweden (Teräsvirta, 2005), Romania (Saman, 2011), and Canada (Tkacz, 2001b). Recently, Torres and Qiu (2018) employed the ANN method to forecast returns from several cryptocurrencies, exchange rates, commodities, and stocks.

Several studies have demonstrated that ANN models yield more accurate predictions than econometric models (Tkacz, 2001a,b; Ali Choudhary and Haider, 2012; McNelis and McAdam, 2004). Shi et al. (2006) used a neural network with a genetic algorithm to forecast China's GDP. The quarterly data employed in the model yielded accurate and efficient estimates. Jahn (2020) demonstrated that an ANN model yielded better performance than a linear model in predicting the annual GDP of 15 industrialized economies. Furthermore, Chuku et al. (2019) estimated the GDP for South Africa, Kenya, and Nigeria and found that an ANN model was superior to traditional econometric models and ARIMA. With increased exposure to chaotic influences (political factors, external factors, and commodity prices) among different countries, linear models are becoming less suitable; ANN models are more accurate given their flexibility in modeling (Chuku et al., 2019).

Uncertainty is common in macroeconomic activity, which is a challenge for researchers keen on predicting the future of such indicators in a constantly changing environment. During the current pandemic period, concerns have been raised about a future global economic crisis. The International Monetary Fund (IMF) projects a 4.9% decline in global economic growth for 2020–21: an 8% decline for advanced economies, and 3% and 1% declines for emerging and low-income economies, respectively (International Monetary Fund, 2020). Similar estimates from the OECD indicated a 0.5 to 1.5% decline in global economic growth, while the World Bank and Asian Development Bank predicted 2.1% to 3.9% and 2.3% to 4.8% declines in global economic growth, respectively (RBI, 2020a,b). The increased spread of the pandemic has led many researchers to estimate the economic consequences of the pandemic outbreak. The studies employing annual data are not well suited for this purpose, as they focus on long-term forecasting, whereas policy measures must be devised in the short term. Although a pandemic outbreak usually has both long- and short-term consequences, short-term analysis is more effective from a policymaking perspective. Quantifying the short-term consequences will lead to better decision making to overcome the tragedies associated with the pandemic.

Against this backdrop, we developed a multilayer ANN model to accurately predict quarterly GDP figures for 8 major economies: the United States, Mexico, Germany, Italy, Spain, France, India, and Japan. These countries represent three continents and have experienced a massive onslaught of COVID-19 spread. The main contribution of this paper to the forecasting literature is that it provides a well-calibrated nonlinear model that can accurately predict the impact of a pandemic such as COVID-19. The adaptive model is suitable for analyzing the economic impact of COVID-19. Such predictive models will provide policymakers with a framework to readjust and reinvigorate their economies and can be used for predictions in other countries, as well as the global GDP.

The advantages of the ANN model (Sivanandam and Deepa, 2013) are as follows:

(1) Adaptation potential: The model can learn a pattern or predict a value by adjusting the weights of connections between neurons of different layers.

(2) Self organization: The model organizes itself during the training phase to subsequently perform the desired task.

(3) Real-time operation: After a satisfactory training phase, the model can provide fast prediction and classification performance because of the parallel operations of the artificial neurons of all layers.

(4) Fault tolerance: Because of the massively interconnected parallel network, the ANN performs satisfactorily even after a small amount of damage to the network.

The rest of this paper is organized as follows: Section 2 develops the ANN forecasting model; the simulation procedure of the model is explained in Section 3; the simulation results are presented in Section 4; a discussion of the results is provided in Section 5; finally, the last section concludes with broad policy messages.

2. Development of the Multilayer Artificial Neural Network (MLANN)-based GDP prediction model

Statistical models are inappropriate for prediction when the data are highly nonlinear, uncorrelated, nonstationary, and chaotic, (Teräsvirta, 2006). Nonlinear models such as ANNs are required to circumvent this situation. The MLANN is a multilayered, fully connected, continuously differentiable, nonlinear network that is an appropriate choice to handle the nonlinearity in data (Haykin, 2009). The steps used in the development of the MLANN-based GDP prediction model are shown in Fig. 1.

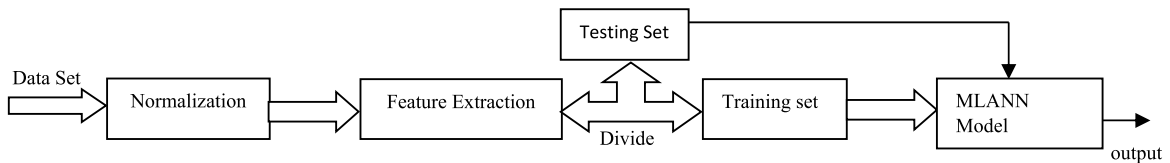


Fig. 1. Steps followed in the development of the GDP prediction model.

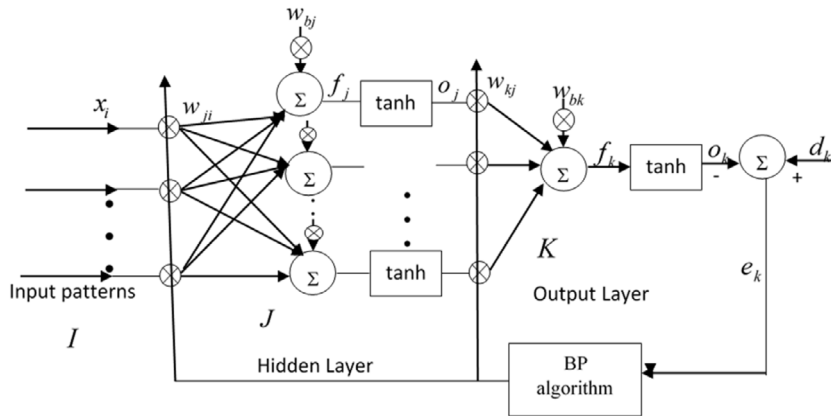


Fig. 2. An MLANN-based GDP prediction model.

The detailed architecture of the MLANN-based prediction model is shown in Fig. 2. Let I, J and K be indices for the input, hidden and output layers, respectively. I represents the number of inputs in each input pattern, J denotes the number of neurons in the hidden layer, and K represents the number of neurons in the output layer. In this case, since the output is one, $K = 1$. Let there be N input patterns: the i th input pattern is represented as x_i . Every input pattern is applied to the input layer of the MLANN model sequentially, weighted, summed, and passed through the activation function (\tanh) to give the output at the hidden layer. The same process continues for the next layer, and the final output of the feedforward network is obtained as o_k . The output is then compared with the desired value or target value, d_k , to calculate the error. This error value is used along with backpropagation (Haykin, 2009) learning to update the weight and bias values of the network. The process continues until the squared error is minimized. The ANN is an adaptive model that iteratively learns from past data during its development stage. Iterative learning means that the connecting weight of each branch updates its old value during the training phase so that the overall training error progressively decreases as the number of iterations increases. The weights are adjusted from their initial random values following a weight update rule known as the back propagation (BP) algorithm. The weight update rules of the BP algorithm are given in Eqs. (6) to (11). The initial weights are chosen randomly; then, the weights of each path of the network are updated using the old weight values, inputs to the weights, back propagated error and learning coefficient. The update process continues until the optimal weights are obtained. The detailed equations for the calculation of the output at each layer and weight update rules are given below.

Referring to the above figure, the output of the k th output neuron o_k is given as

$$o_k = \tanh(f_k) \tag{1}$$

where

$$f_k = \sum_{j=1}^J o_j w_{kj} + w_{bk} \tag{2}$$

o_j = output at the j th hidden neuron

w_{kj} = weights between the j th hidden neuron and k th output neuron

w_{bk} = bias at the k th output neuron

Similarly, the output at the j th hidden layer, o_j , is calculated as

$$o_j = \tanh(f_j) \tag{3}$$

where

$$f_j = \sum_{i=1}^I x_i w_{ji} + w_{bj} \tag{4}$$

x_i = *ith input pattern*
 w_{ji} = *weights between ith input and jth hidden neuron*
 w_{bj} = *bias at jth hidden neuron*

The output of the prediction model o_k is compared with the corresponding target value d_k to determine the error. Hence,

$$e_k = d_k - o_k \tag{5}$$

The weights between the hidden layer and output layer, w_{kj} , are updated using

$$w_{kj} = w_{kj} + \mu * \delta_k * o_j \tag{6}$$

where

$$\delta_k = e_k * \frac{(1 - o_k^2)}{2} \tag{7}$$

μ = *learning parameter lies between 0 to 1*

The bias weight, whose input is always one, is updated as

$$w_{bk} = w_{bk} + \mu * \delta_k \tag{8}$$

Similarly, the weights between the input layer and the hidden layer, w_{ji} , are updated using

$$w_{ji} = w_{ji} + \mu * \delta_j * x_i \tag{9}$$

where

$$\delta_j = \delta_k * w_{kj} * \frac{(1 - o_j^2)}{2} \tag{10}$$

The bias weight of the *jth* neuron in the hidden layer is updated as

$$w_{bj} = w_{bj} + \mu * \delta_j \tag{11}$$

Eqs. (1)–(11) are the key equations for developing the MLANN-based GDP prediction model.

3. Simulation procedure

The simulation of the MLANN-based GDP prediction model is conducted referring to Fig. 2 and using MATLAB 2016 software. The steps are explained in the following section.

(i) Data collection and normalization: Quarterly GDP data were collected for Japan, China, Germany, Spain, France, Italy, the USA, and Mexico from <https://fred.stlouisfed.org>, Economic Research Division, Federal Reserve Bank of St. Louis (FRED | St. Louis Fed, 2020). For India, the data were collected from the Reserve Bank of India (RBI, 2020a,b). The time period of the data and details about the data are provided in Table 1. The GDP data are normalized to a range of 0–1 by dividing each observation by the maximum value of the distribution. Large differences between observations in the data cause problems during modeling calibration, and normalized data help overcome this problem during the calibration of the model and increase the convergence speed.

(ii) Feature extraction: Normalization of the data is followed by feature extraction using a sliding window of size three to generate data patterns or feature patterns. The window is moved over the entire data with a shift of one, and each time, a group of three values is obtained. If N is the length of the data, then there are $N - 2$ groups. For the n th group, $x(n)$, $x(n + 1)$, $x(n + 2)$ are the available values. Every data or input pattern consists of five values, i.e., three values of the group and the slopes between the 1st and 3rd values and 2nd and 3rd values. Mathematically, these values can be represented as: $\{x(n), x(n + 1), x(n + 2), (x(n + 2) - x(n + 1))/x(n + 2), (x(n + 2) - x(n))/x(n + 2)\}$. Hence, at each time point, five inputs are fed to the MLANN model in one input pattern, and there are $N - 2$ patterns in total. Since MLANN is a supervised learning-based model, the target value is also known and stored. For the n th input data pattern, $x(n + 3)$ is the required desired value or target value. The total number of patterns generated for each of the datasets is given in Table 1. Of the total data patterns or input patterns generated, 80% are used to train the model, and the remaining 20% are used to test the model.

(iii) Training of the model: The development of an MLANN-based GDP prediction model is formulated as an optimization problem, where the error between the target value and model estimated values is minimized towards zero. Once the error is nearly zero, the model can predict the correct value of the GDP. A 9:3:1 MLANN structure is used in this paper to

Table 1
Details of the data.

Sl. no.	Name of country	Length of time-series	Length of data	Total number of patterns extracted	Total no. of training patterns	Total number of testing patterns
1	USA	01/01/1990 to 01/10/2020	121	119	97	22
2	Mexico	01/01/1993 to 01/01/2020	109	107	86	21
3	Italy	01/01/1995 to 01/01/2020	101	99	79	20
4	Germany	01/01/1991 to 01/01/2020	117	115	92	23
5	Spain	01/01/1995 to 01/01/2020	101	99	79	20
6	France	01/01/1990 to 01/01/2020	121	119	95	24
7	Japan	01/01/1994 to 01/01/2020	105	103	82	21
8	India	01/04/2011 to 31/03/2020	36	34	28	06

predict GDP. The model has two hidden layers with 9 and 3 neurons and an output layer with one neuron. Initially, all the connecting weights and bias values are randomly selected between -0.5 and 0.5 . The choice of initial values affects the convergence speed and the final mean square error after the training phase. To make the training process unbiased, the initial weights are randomly selected. The final weights after the training process may have positive or negative values; therefore, the initial weights are chosen to be both negative and positive values. Further, to achieve unbiased selection, the weights are chosen from a uniform distribution with zero mean (unbiased) ranging between -0.5 and 0.5 . The total numbers of weights required between the input layer and first hidden layer, first hidden layer and second hidden layer, and second hidden layer and output layer are 5×9 , 9×3 , and 3×1 , respectively. Similarly, the numbers of biases are 9, 3, and 1 for the first hidden, second hidden, and output layers, respectively. The first data pattern with five values is fed to the model, and the input is weighted and passed through an activation function to produce an output at the first hidden layer. The same process is repeated for the second hidden layer and finally the output layer. The output is then compared with the corresponding target value to calculate the error. The BP rule is used to update the weights and bias values using Eqs. (6)–(11). In the same way, all input patterns are fed to the model sequentially, and the weights are updated until all input patterns are exhausted. This process completes one experiment and is repeated 50,000 times in 10 independent runs. The value of the learning parameter is 0.1. During every experiment, the mean squared error (MSE) is calculated and plotted to illustrate the convergence characteristics of the model. The error convergence plots during training are shown in Figs. 4(a)–(h), and a comparison of the actual and predicted values of the model during training is shown in Figs. 3(a)–(h). Once the error is minimized, the training process is stopped, and the final values of the weights and biases are saved for testing purposes.

(iv) Testing or validation of the model: Testing or validation of the model is performed using the 20% of the input or data patterns that are not used during training. The testing patterns are input into the trained model sequentially, and the output is obtained after weighting, adding, and passing through the activation function. Each of the outputs of the model is compared with the available target value to calculate the mean absolute percentage error (MAPE) using (12)

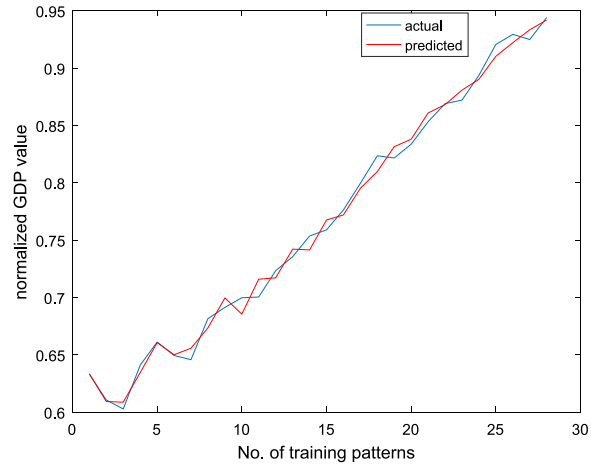
$$MAPE = \frac{1}{L} \sum_{l=1}^L \text{abs}((a(l) - p(l))/a(l)) \times 100 \quad (12)$$

where $a(l)$ = actual value of the l th testing pattern
 $p(l)$ = predicted value of the l th testing pattern

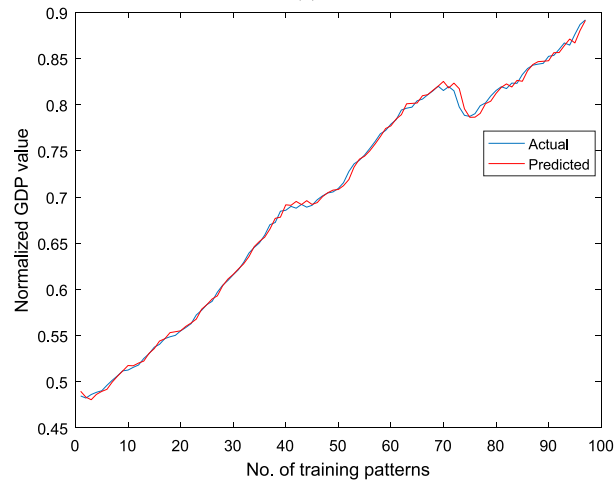
4. Simulation results

Figs. 3(a) to 3(h) present a comparison of the actual data and the forecasted estimates in the training process. The model has been trained accurately to capture the spikes present in the data.

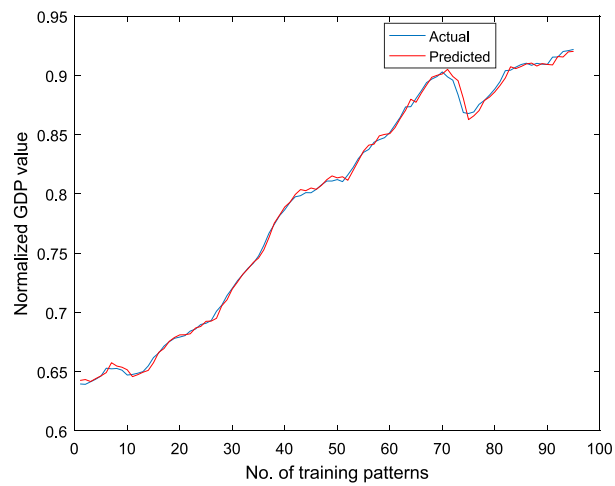
Figs. 4(a) to 4(h) show the MSE of the estimated GDP outputs obtained from the training of the model. The figures show that the MSE curve starts with a high value but declines with each iteration and finally becomes parallel to the X-axis, showing that it has reached its minimum. Additional iterations do not further reduce the MSE, and the weights obtained from the model are optimum.



3(a) India

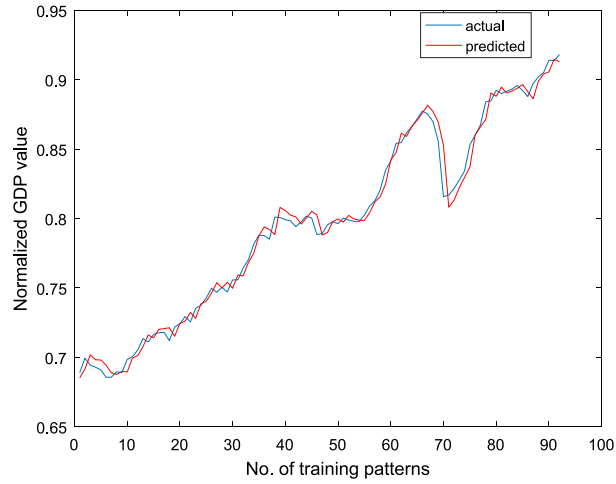


3(b) United States

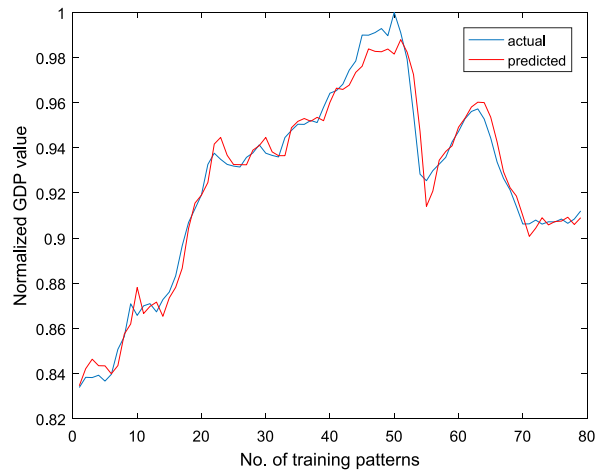


3(c) France

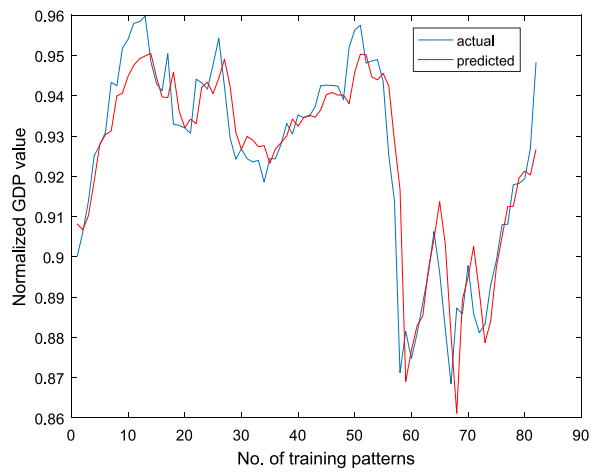
Fig. 3. Comparison of Actual and Predicted GDP during Training of the Model.



3(d) Germany

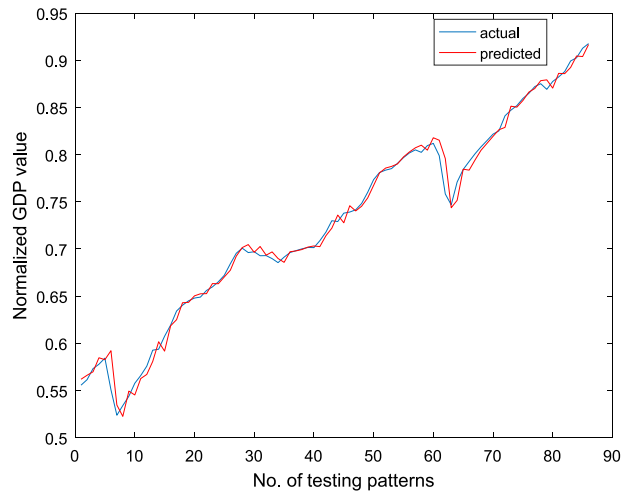


3(e) Italy

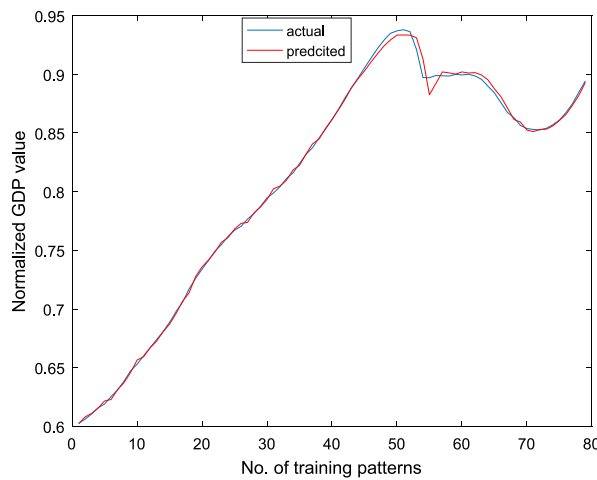


3(f) Japan

Fig. 3. (continued).



3(g) Mexico



3(h) Spain

Fig. 3. (continued).

Table 2 below shows the MAPE of the predicted GDP values during testing of the MLANN model for the 8 countries. The MAPE is an indicator of how close the predicted values are to the actual values. In all countries, the MAPE is 2% or less, which suggests that the ANN model developed to predict future values is well calibrated. The last column of Table 2 presents the forecasted GDP figures for the 8 countries considered in this study.

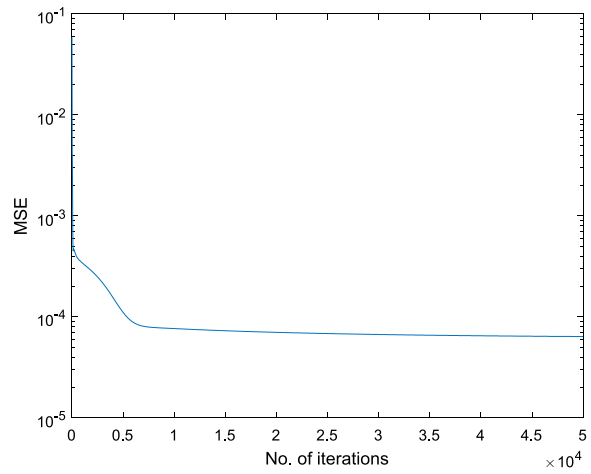
A comparison of the actual and predicted values of quarterly GDP for the countries during the testing of the model is provided in Appendices A and B.

5. Discussion of the results

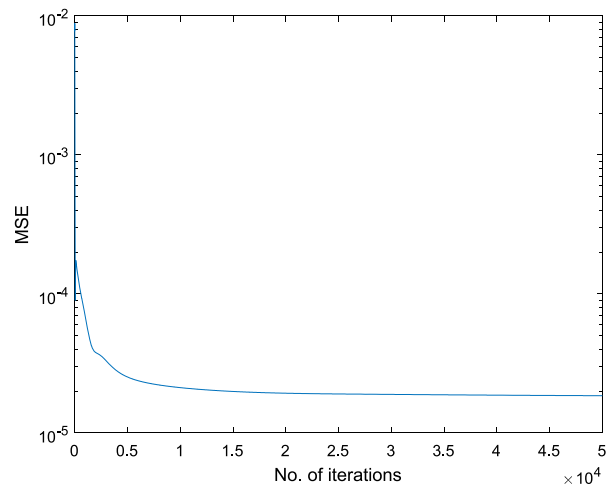
Table 3 presents the quarter-to-quarter growth rate for Q1 (January–March) and Q2 (April–June) for the 8 major economies. The Q1 growth rate is estimated using actual data, whereas the Q2 growth rate is based on the percentage change between the forecasted GDP figure of Q2 and the actual figure for Q1. Furthermore, the annual growth rate for all countries is given in the last column of Table 3. The quarter-to-quarter and annual growth rates are based on the following formula:

$$\text{Quarter GDP Growth } (G_q) = \frac{GDP_q - GDP_{q-1}}{GDP_{q-1}} \tag{13}$$

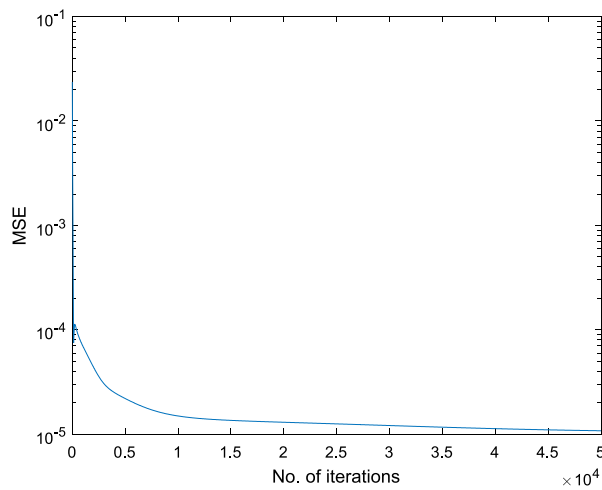
$$\text{GDP Annual } (G_a) = (1 + G_q)^4 - 1 \tag{14}$$



4(a) India

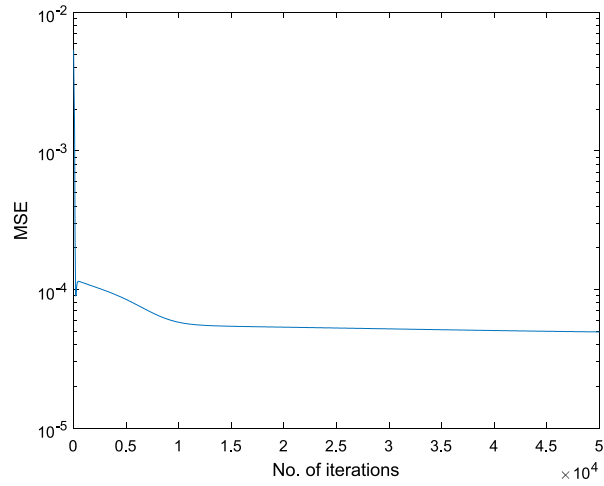


4(b) United States

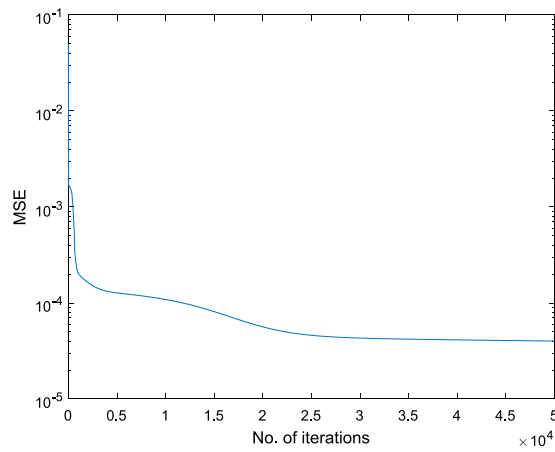


4(c) France

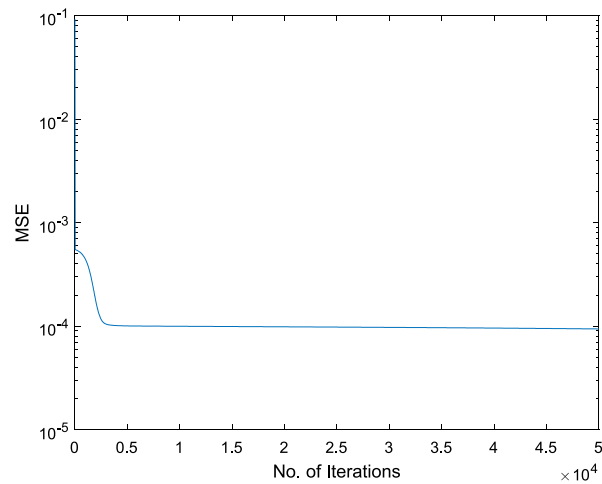
Fig. 4. MSE of Predicted GDP during Training of the Model.



4(d) Germany

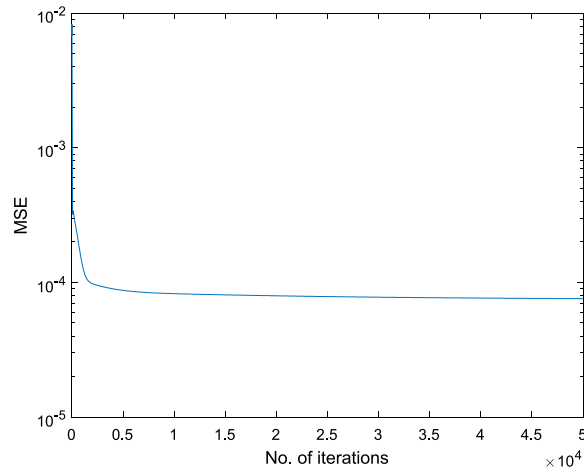


4(e) Italy

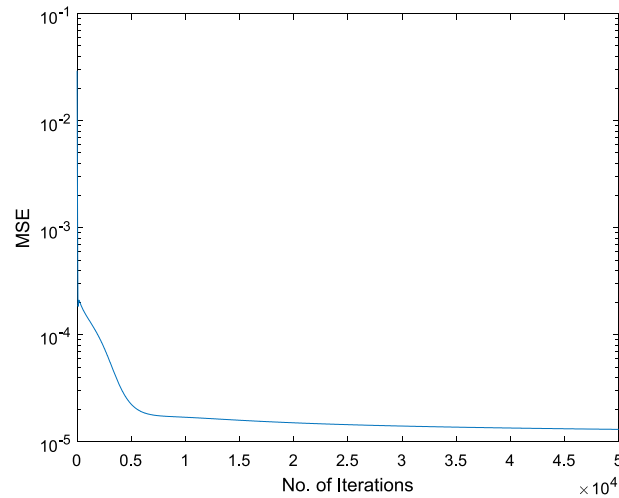


4(f) Japan

Fig. 4. (continued).



4(g) Mexico



4(h) Spain

Fig. 4. (continued).

Table 2
Mean absolute percentage error (MAPE) of GDP prediction during testing.

Country	MAPE during testing	Forecasted GDP value for April–June quarter (in local currency)
India	1.2503	33026
US	1.4458	18454
Germany	1.9925	724250
Japan	2.0429	533590
Italy	0.4479	373980
Spain	1.5360	274840
Mexico	1.9523	4450
France	1.5953	522350

Note: The forecasted GDPs of seven countries are represented in their respective currency – India in billion Rupee, United States in billion dollars, Germany, Spain, France and Italy in million Euro, Japan in billion Yen, Mexico in billion pesos.

The extreme severity of COVID-19 has prompted several governments to take necessary precautions based on the spread of infections. These measures include localized recommendations, national recommendations, and regional and

Table 3
Quarterly growth rates during COVID-19 shutdown.

Country	Quarterly growth (Jan–Mar, 2020) based on actual data	Quarterly growth (Apr–Jun, 2020) based on forecasted figure	Annual growth rate
United States	–1.29	–2.74	–10.53
Mexico	–1.60	–2.14	–8.30
France	–5.34	–1.78	–6.95
Spain	–5.24	–2.72	–10.45
Italy	–5.31	–2.40	–9.26
Germany	–2.22	–1.04	–4.11
Japan	–0.49	–2.42	–9.35
India	2.38	–2.78	–10.67

Note: Quarterly growth rate represents present quarter to previous quarter growth. Annual growth rate indicates the growth estimated at an annual rate, including all four quarters.

national lockdowns. As the disease containment measures became increasingly stringent by mid-March 2020, the disruption of supply chains and slowing of economic activities became drastic. However, a significant decline in worldwide economic growth was already noted even before COVID-19, mainly due to the trade war between China and the United States and the steep decline in consumer expenditure ([World Economic Outlook Update, 2019](#)).

[Table 3](#) shows that most economies shrank in the 1st quarter of 2020. The United States, the largest economy in the world, experienced a 1.3% decline in GDP growth in the January to April quarter when compared to the previous quarter. The US economy initiated regional restrictions on movement based on the severity of the pandemic in mid-March. However, the first quarter captured only 2 weeks of the lockdown, and the impact loomed large in the second quarter due to increased pandemic outbreaks and restrictions in different states of the country. Mexico experienced a situation similar to that of the US, where restrictions were initiated in mid-March and GDP growth fell by 1.6% in the first quarter of 2020.

While North American countries restricted movements to a certain extent to curb the spread of the disease, European countries, such as Italy, Spain, Germany, and France – the four largest Euro economies that had the highest number of infections – imposed a stricter form of lockdown. The strict lockdown measures caused economic contractions of 5.34%, 5.24%, and 5.31% in France, Spain, and Italy, respectively, in the first quarter of 2020. The largest economy of the Eurozone, Germany, experienced a 2.2% GDP contraction during the same time. Although flattening of the pandemic outbreak curve was achieved in European countries by the end of June, the second quarter of 2020 may entail a steep decline in GDP due to the complete shutdown of economic activities.

The greatest impact is observed in countries such as France and Spain, which have a strong dependence on tourism and the service industry compared to Germany. Asian countries showed modest growth from January to March. India's GDP grew at 2.38% in this quarter compared to the previous quarter. India imposed a nationwide lockdown on the 25th of March 2020. Hence, the January to March growth rate does not reflect the impact of the lockdown; rather, it reflects the already slowing economy even before the outbreak of COVID-19. Surprisingly, air pollution, as a byproduct of economic activity, has decreased during this period ([Kumar and Managi, 2020](#)). The Japanese economy contracted by 0.5% from January to March compared to the previous quarter, the second straight decline in economic growth for two consecutive quarters. Reduced exports amid a trade war between the US and China, followed by a slump in consumer expenditure, have had a severe impact on the Japanese economy.

The devastating spread of the pandemic and extension of restrictions among several countries has resulted in a steep decline in economic growth and led to recession ([IMF, 2020](#)). The decline in aggregate demand caused increased layoffs, leading to higher unemployment. The key policies by the central banks of different countries to lower the impact and induce liquidity are evident. The world economy was on the cliff, with significant events such as geopolitical tensions and trade wars, and the coronavirus pushed the economy over the edge ([World Bank, 2020](#); [United Nations, 2020](#)).

We forecast the GDP growth for the 2nd quarter of 2020 because of the policy implications. The forecasted GDP for the April to June quarter and the annualized GDP growth indicate a steep decline among the major economies of the world. The forecasted GDP growth rates in the United States and Mexico are –2.74% and –2.4%, respectively, for the 2nd quarter. The restrictions in these countries have disrupted the movement of several essential and nonessential goods. However, what is more concerning is the double-digit decline in annualized GDP growth, which stands at –10.53% and –8.3% in the US and Mexico, respectively. The decreased demand for petroleum products at home and export goods abroad might have caused such a substantial negative impact on the US economy. Furthermore, increased unemployment and severe disruption of service sector-related industries, such as hotels and airlines, may lead to a sharp decline in economic growth and personal consumption.

The disruption in the Eurozone is expected take quite a long time to recover, as most of the member countries are heavily dependent on tourism and services. The major economies of the Eurozone – France, Spain, Italy, and Germany – are predicted to contract by 1.78, 2.72, 2.4, and 1.04%, respectively. The unemployment stimulus package in the Euro area is smaller than that in the United States due to the widespread use of short-term work policies ([World Bank, 2020](#)). Among Asian countries, Japan prevented the spread of the pandemic by means of national recommendations. Although the

Table 4
Comparison of GDP annual growth forecasts by our model and other leading global agencies.

Country	Our model	IMF	World Bank	Morgan Stanley
United States	−10.53	−8	−7.9	−5.8
Mexico	−8.30	−10.5	−8.7	
France	−6.95	−12.5		
Spain	−10.45	−12.8		
Germany	−4.11	−7.8	−10.1	−9.6
Italy	−9.26	−12.8		
Japan	−9.35	−5.8	−6.8	
India	−10.67	−4.5	−9	−1.7

Note: The GDP growth rates estimated by various financial agencies were taken from different reports for comparison purposes.

Japanese economy was opened for trade, a strong contraction in GDP is observed in the second quarter, which follows the decline experienced from October to December 2019. The spillover effect of the trade war and decline in global aggregate demand may continue, which would further hinder the growth of the Japanese economy. While predicting Japan's GDP in the training dataset (see Fig. 3f), a significant gap was observed between the actual and predicted values for some of the previous quarters. One explanation of this gap is that the predictive accuracy of a model is affected when an economy faces frequent uncertain events. This is what occurred for the Japanese economy, as it has been faced with several uncertain policy decisions resulting from an unsustainable fiscal trajectory, constraints on monetary policy, uncertainty around world trade policies, and weak growth (Arbatli et al., 2019). A difference between the actual and predicted values of GDP is also observed for Italy (see Fig. 3e). This prediction error may be explained by the prolonged period of policy uncertainty in both countries.

In India, a nationwide lockdown was initiated in the last week of March and was continued in the subsequent months of the next quarter with localized lockdown and nationwide recommendations. The forecasted growth rate of GDP in India is negative, i.e., -2.78% , and the annualized growth rate for 2020–21 is predicted to be -10.67% . Towards the end of the April–June quarter, the pandemic was raging through rural areas of the country as the migrant reverse exodus occurred, in which millions of migrant workers returned to their native rural homes from the cities. At the time of writing this paper, the pandemic had affected 1.7 million lives in India, and several states were in and out of lockdown. Therefore, a significant contraction in GDP growth in India, as forecasted by us, is not surprising. While the fiscal stimulus provided by different central banks would reduce the contraction in the 2nd quarter, the overall economic outlook in most of the countries in 2020 looks bleak.

In Table 4, the forecasts from the ANN model are compared with the forecasts from leading global agencies to enhance the understanding of the world economic outlook in 2020. Morgan Stanley's outlook for 2020 is optimistic, with a V-shaped recovery, and their forecasts show a modest decline in GDP for the 8 countries we considered. Their optimism is based on the slender recovery in economic activities observed in May. The projections by the World Bank and IMF are closer to our forecasts. The forecasted decline in GDP growth for all these countries warrants a strong response from governments to prevent economies from slipping into recession.

Although past economic shocks have contracted the world economy, the global financial crisis of 2008 was the greatest slump since the great depression in 1930. Low-interest rate policies, overleveraging, and unsustainable fiscal and monetary policies led to a global financial crisis, causing a credit crunch and unemployment, which pushed world economies into a deep recession. World GDP growth slowed from 5% in 2007 to 3.75% in 2008 and 2% in 2009. However, with remedial measures, such as quantitative easing, financial reforms by the governments and central banks, signs of recovery were evident in 2010. Other past pandemic outbreaks had negative effects on economies, but they were limited to a few countries. The 1918 Spanish influenza had a significant impact on major economies. The projected reduction in quarterly GDP was 2.6% for the United States (Dixon et al., 2010), and strong evidence of a negative effect on capital returns was found in Sweden (Karlsson et al., 2014). The short-term impact of Ebola on most African economies was negative, as GDP growth fell by 2.1% in Guinea, 3.4% in Liberia, and 3.3% in Sierra Leone (World Bank, 2014). The specific countries facing pandemics in the past had a small impact on the world economy, while the COVID-19 pandemic has had a widespread negative impact on global economic growth and trade. The ongoing trade war and uncertain worldwide events, followed by the pandemic outbreak, have led to an unprecedented economic crisis. This uncertain nature of the economic impact of the pandemic has resulted in a gap between the actual and forecasted quarterly GDP growth during the COVID-19 shutdown period. The quarter-to-quarter fluctuations in GDP are the reason for the forecasting error during the shutdown period. The current model, which has accurately predicted the GDP during the training period, needs improvement to capture the large fluctuations.

6. Conclusion

The main contribution of this paper is the development of an ANN model to forecast GDP one quarter ahead for eight major economies. This model captures the nonlinearities present in the quarterly time-series data and provides accurate

Table A.1
Actual and predicted value obtained during testing pattern.

United States		Germany		Japan		Italy	
1	2	1	2	1	2	1	2
1.714	1.713	6.880	6.843	5.293	5.180	3.885	3.873
1.728	1.720	6.939	6.869	5.317	5.262	3.894	3.896
1.741	1.732	6.925	6.927	5.329	5.284	3.913	3.905
1.746	1.743	6.968	6.902	5.316	5.279	3.926	3.928
1.747	1.747	7.000	6.938	5.353	5.270	3.934	3.941
1.756	1.747	7.030	6.967	5.344	5.288	3.955	3.945
1.764	1.755	7.086	6.987	5.353	5.289	3.967	3.971
1.774	1.762	7.127	7.031	5.369	5.288	3.990	3.983
1.782	1.770	7.142	7.058	5.407	5.298	4.004	4.008
1.793	1.777	7.167	7.060	5.432	5.318	4.019	4.022
1.802	1.784	7.253	7.075	5.491	5.331	4.041	4.035
1.816	1.791	7.293	7.139	5.510	5.355	4.041	4.057
1.832	1.801	7.356	7.160	5.475	5.366	4.041	4.051
1.844	1.812	7.410	7.191	5.490	5.348	4.037	4.047
1.860	1.819	7.420	7.218	5.455	5.350	4.044	4.043
1.873	1.829	7.448	7.215	5.463	5.339	4.052	4.051
1.878	1.836	7.441	7.227	5.525	5.338	4.055	4.062
1.893	1.838	7.456	7.220	5.559	5.367	4.056	4.063
1.902	1.846	7.491	7.227	5.581	5.384	4.047	4.062
1.912	1.850	7.473	7.247	5.495	5.390	3.832	4.049
1.922	1.854	7.493	7.233	5.469	5.357		
1.898	1.859						

Note: The table includes the total number of testing pattern and obtained (1) Actual values and (2) Predicted Values for different countries.

predictions. These countries have experienced substantial negative health impacts from the ongoing COVID-19 pandemic. The infection and fatality rates have been alarming, and governments have implemented various forms of lockdown to contain the disease. As a result, their economies have been disrupted, resulting in the shutting down of many industries and rising unemployment rates. In such a situation, a clear picture of what lays ahead in terms of economic outlook will help policymakers take necessary steps. For example, opinion has been divided in the US Congress as to how much relief is sufficient to restart the economy. The ANN model developed in our paper accurately predicted the GDP figures, as the MAPE is less than 2% in each of the country cases.

The findings show that the April to June quarters of 2020 will see a significant decline in economic growth in all eight countries. The annualized GDP growth shows an even larger impact, as most countries will experience double-digit negative economic growth. Such a scenario is expected, though it requires strong corrective actions by central banks and governments. The US government has announced a rescue package of \$2.2 trillion ([The Hindu, 2020](#)), which is the largest rescue package in recent decades. Joseph Stiglitz, the Nobel laureate in Economic Sciences, in an interview with the British tabloid *The Independent* has opined that an even larger rescue package to the tune of \$6 trillion (a third of the US GDP) may be required. Similarly, the Indian government declared a \$260 billion coronavirus rescue package ([The New York Times, 2020](#)). This rescue package is supposed to support small- and medium-scale industries and the agriculture sector, reinstate migrant laborers, and support ailing banks and financial institutions. Furthermore, most central banks have injected liquidity into the economy by reducing interest rates.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix A

See [Table A.1](#).

Appendix B

See [Table B.1](#).

Table B.1
Actual and predicted value obtained during testing pattern.

France		Mexico		Spain	
1	2	1	2	1	2
5.184	5.175	4.265	4.261	2.667	2.663
5.208	5.178	4.309	4.268	2.691	2.689
5.215	5.201	4.362	4.313	2.717	2.708
5.242	5.205	4.361	4.350	2.736	2.729
5.239	5.227	4.387	4.336	2.747	2.742
5.256	5.221	4.407	4.370	2.771	2.746
5.262	5.235	4.455	4.377	2.784	2.768
5.300	5.239	4.506	4.415	2.805	2.776
5.287	5.271	4.527	4.441	2.834	2.791
5.299	5.256	4.542	4.449	2.849	2.814
5.329	5.264	4.528	4.459	2.869	2.821
5.374	5.290	4.581	4.448	2.883	2.832
5.411	5.323	4.642	4.488	2.899	2.840
5.450	5.346	4.630	4.509	2.913	2.848
5.495	5.368	4.645	4.494	2.930	2.856
5.504	5.393	4.648	4.511	2.946	2.864
5.515	5.394	4.642	4.508	2.957	2.872
5.534	5.398	4.637	4.506	2.969	2.877
5.569	5.408	4.633	4.505	2.982	2.882
5.598	5.427	4.627	4.503	2.825	2.887
5.612	5.440	4.553	4.500		
5.624	5.445				
5.619	5.449				
5.318	5.445				

Note: The table includes the total number of testing pattern and obtained (1) Actual values and (2) Predicted Values for different countries.

References

- Acemoglu, D., Chernozhukov, V., Werning, I., Whinston, M., 2020. A multi-risk SIR model with optimally targeted lockdown. *Natl. Bur. Econ. Res.* <http://dx.doi.org/10.3386/w27102>.
- Ali Choudhary, M., Haider, A., 2012. Neural network models for inflation forecasting: An appraisal. *Appl. Econ.* 44, 2631–2635. <http://dx.doi.org/10.1080/00036846.2011.566190>.
- Arbatli, E.C., Davis, A., Ito, S.J., Miale, N., 2019. Policy Uncertainty In Japan. NBER WORKING PAPER SERIES, Working Paper 23411. <http://www.nber.org/papers/w23411>.
- Balcilar, M., Gupta, R., Majumdar, A., Miller, S.M., 2015. Was the recent downturn in US real GDP predictable? *Appl. Econ.* 47, 2985–3007. <http://dx.doi.org/10.1080/00036846.2015.1011317>.
- Bañbura, M., Giannone, D., Reichlin, L., 2010. Large Bayesian vector auto regressions. *J. Appl. Econom.* 25, 71–92. <http://dx.doi.org/10.1002/jae.1137>.
- Box, G., Jenkins, G., Reinsel, G., Ljung, G., 2015. *Time series analysis: forecasting and control*.
- Chuku, C., Simpasa, A., Oduor, J., 2019. Intelligent forecasting of economic growth for developing economies. *Int. Econ.* 159, 74–93. <http://dx.doi.org/10.1016/j.inteco.2019.06.001>.
- Dixon, P.B., Lee, B., Muehlenbeck, T., Rimmer, M.T., Rose, A.Z., Verikios, G., 2010. Effects on the U.S. of an H1N1 epidemic: analysis with a quarterly CGE model. Centre of Policy Studies/IMPACT Centre Working Papers g-202, Victoria University, Centre of Policy Studies/IMPACT Centre.
- FRED | St. Louis Fed, 2020. Economic research division, Federal Reserve Bank of St. Louis. <https://fred.stlouisfed.org>.
- Guégan, D., Rakotomaroahy, P., 2010. Alternative methods for forecasting GDP. *Int. Symp. Econ. Theory Econ.* 20, 161–185. [http://dx.doi.org/10.1108/S1571-0386\(2010\)0000020013](http://dx.doi.org/10.1108/S1571-0386(2010)0000020013).
- Härdle, W., Müller, M., Sperlich, S., Werwatz, A., 2012. Nonparametric and semiparametric models. http://dx.doi.org/10.1007/978-3-642-17146-8_1.
- Haykin, Simon, 2009. *Neural networks and learning machines (3rd edition)* by Simon Haykin. Neural Netw..
- Hlaváček, H.H., Koňák, M., Koňák, K., Michael, 2005. *The Application of Structured Feedforward Neural Networks To the Modelling of Daily Series of Currency in Circulation*. Springer.
- IMF, 2020. World Economic Outlook Update: A Crisis Like No Other, An Uncertain Recovery, World Economic Outlook Update.
- International Monetary Fund, 2020. World economic outlook, January 2020, tentative stabilization, sluggish recovery? *World Econ. Outlook* 1–10.
- Jahn, M., 2020. Artificial neural network regression models in a panel setting: Predicting economic growth. *Econ. Model.* 91, 148–154. <http://dx.doi.org/10.1016/j.econmod.2020.06.008>.
- Jena, P.R., Majhi, R., Majhi, B., 2015. Development and performance evaluation of a novel knowledge guided artificial neural network (KGANN) model for exchange rate prediction. *J. King Saud. Univ. - Comput. Inf. Sci.* 27, 450–457. <http://dx.doi.org/10.1016/j.jksuci.2015.01.002>.
- Karlsson, M., Nilsson, T., Pichler, S., 2014. The impact of the 1918 Spanish flu epidemic on economic performance in Sweden: an investigation into the consequences of an extraordinary mortality shock. *J. Health Econ.* 36, 1–19.
- Kock, A.B., Teräsvirta, T., 2014. Forecasting performances of three automated modelling techniques during the economic crisis 2007–2009. *Int. J. Forecast.* 30, 616–631. <http://dx.doi.org/10.1016/j.ijforecast.2013.01.003>.
- Kumar, S., Managi, S., 2020. Does stringency of lockdown affect air quality? Evidence from Indian cities. *Econ. Disasters Clim. Change* 4, 481–502.
- Loermann, J., Maas, B., 2019. Munich Personal RePEc Archive Nowcasting US GDP with artificial neural networks.
- McNelis, P.D., 2005. Neural networks in finance: gaining predictive edge in the market. <http://dx.doi.org/10.1017/CBO9781107415324.004>.
- McNelis, P.D., McAdam, P., 2004. *Forecasting inflation with thick models and neural networks*.
- Nakamura, H., Managi, S., 2020. Airport risk of importation and exportation of the COVID-19 pandemic. *Transp. Policy* 96, 40–47. <http://dx.doi.org/10.1016/j.tranpol.2020.06.018>.

- Ramos-Pérez, E., Alonso-González, P.J., Núñez Velázquez, J.J., 2019. Forecasting volatility with a stacked model based on a hybridized artificial neural network. *Expert Syst. Appl.* 129, 1–9. <http://dx.doi.org/10.1016/j.eswa.2019.03.046>.
- RBI, 2020a. Reserve bank of India. Monetary Policy Report April 2020.
- RBI, 2020b. Reserve bank of India. In: *HandBook of Statistics on Indian Economy*.
- Saman, C., 2011. Scenarios of the Romanian Gdp evolution with neural models 1. *Romanian J. Econ. Forecast.* 4.
- Shi, P., Chen, Z., Xie, G., 2006. Using artificial neural network trained with genetic algorithm to model GDP prediction.
- Sivanandam, S.N., Deepa, S.N., 2013. *Principles of Soft Computing*, second ed. Wiley.
- Szafranek, K., 2019. Bagged neural networks for forecasting polish (low) inflation. *Int. J. Forecast.* 35, 1042–1059. <http://dx.doi.org/10.1016/j.ijforecast.2019.04.007>.
- Teräsvirta, T., 2005. *A Service of Zbw Standard-Nutzungsbedingungen: Forecasting Economic Variables with Nonlinear Models*. Elsevier.
- Teräsvirta, T., 2006. Chapter 8 forecasting economic variables with nonlinear models. *Handb. Econ. Forecast.* 1, 413–457. [http://dx.doi.org/10.1016/S1574-0706\(05\)01008-6](http://dx.doi.org/10.1016/S1574-0706(05)01008-6).
- The Hindu, 2020. U.S. Senate unanimously passes \$2.2 trillion coronavirus rescue package. Accessed at <https://www.thehindu.com/news/international/us-senate-unanimously-passes-22-trillion-coronavirus-rescue-package/article31169709.ece>, on 1st August, 2020. (Accessed 8 August 20).
- The New York Times, 2020. Modi announces \$260 billion coronavirus rescue package for India. Accessed at <https://www.nytimes.com/2020/05/12/world/asia/virus-india-rescue-package.html>. (Accessed 1 August 2020).
- Tkacz, G., 2001a. Neural network forecasting of Canadian GDP growth. *Int. J. Forecast.*
- Tkacz, G., 2001b. Neural network forecasting of Canadian GDP growth. *Int. J. Forecast.* 17, 57–69. [http://dx.doi.org/10.1016/S0169-2070\(00\)00063-7](http://dx.doi.org/10.1016/S0169-2070(00)00063-7).
- Torres, D.G., Qiu, H., 2018. Applying Recurrent Neural Networks for Multivariate Time Series Forecasting of Volatile Financial Data. WHO.
- United Nations, New York, 2020. *World Economic Situation and Prospects*.
- WHO, 2020. *Monitoring Health for the SDGs*. WHO.
- World Bank, 2020. *World Bank Report. Global Economic Prospects*.
- World Economic Outlook Update, 2019. *A weakening global expansion*. (Accessed 8 August 20).
- Yoo, S., Managi, S., 2020. Global mortality benefits of COVID-19 action. *Technol. Forecast. Soc. Change* 160, 120231. <http://dx.doi.org/10.1016/j.techfore.2020.120231>.

A comparative study of antibacterial and antifungal activities of extracts from four indigenous plants

Rajendra Mehta¹, Suraj Dhruv¹, Vidyanshu Kaushik¹, Kamal Kumar Sen¹, Naureen Shaba Khan², Amar Abhishek³, Ashwini Kumar Dixit³ & Vibhay Nath Tripathi^{3,*}

¹Department of Rural Technology, Guru Ghasidas University, Bilaspur, India 495009; ²Department of Botany, Dr. C. V. Raman University, Bilaspur, India 495113; ³Department of Botany, Guru Ghasidas University, Bilaspur, India 495009; Vibhay Nath Tripathi - Phone: +91-7587879195; E-mail: vibhay.tripathi@ggu.ac.in; *Corresponding author.

Author e-mail contacts: Vibhay Nath Tripathi - vibhay.tripathi@ggu.ac.in; Rajendra Mehta- rajendra64@ggu.ac.in; Naureen Saba Khan - sabanaurinekhan@cvru.ac.in

Received February 19, 2020; Revised March 5, 2020, Accepted March 17, 2020; Published March 31, 2020

DOI: 10.6026/97320630016267

Declaration on Publication Ethics:

The authors state that they adhere with COPE guidelines on publishing ethics as described elsewhere at <https://publicationethics.org/>. The authors also undertake that they are not associated with any other third party (governmental or non-governmental agencies) linking with any form of unethical issues connecting to this publication. The authors also declare that they are not withholding any information that is misleading to the publisher in regard to this article.

Abstract:

A repetitive and wide use of chemical antibiotics has brought a serious threat in the biomedical and clinical sectors by the emergence of multidrug resistant pathogens. Plants have secondary metabolites that make them suitable candidate for natural antimicrobial agent without any side effect. In this study, we assessed comparative antibacterial and antifungal effects of extracts from four Indigenous plants (*Nerium* sp; *Mentha* sp; *Aloe vera* and *Eucalyptus* sp). Total phenolic and flavonoid content were extracted by microwave-assisted extractor and used for phytochemical assay. Antimicrobial experiment was done by micro dilution technique. A post hoc analysis inbuilt with one-way ANOVA test was used for the compilation of antibiotic sensitivity data and percent inhibition. Total phenolic content was significantly high in *Mentha* sp. and low in *Nerium* sp. (All $p < 0.05$). In antibacterial and antifungal activity higher concentrations of extracts showed a strong activity, which was as good as antibiotics used as control. Results from *Eucalyptus* sample showed a significant growth reducing capability even at lower concentrations. This study concludes that the plant extracts can be used to treat microbial infections with almost same efficacy as antibiotics and with a lower chance of resistance development.

Keywords: Antibiotic sensitivity; *Mentha* spp; *Aloe vera*; *Nerium oleander* and *Eucalyptus* spp

RESEARCH ARTICLE



BENTHAM
SCIENCE

Gallic Acid Protects from Acute Multiorgan Injury Induced by Lipopolysaccharide and D-galactosamine



Samrat Rakshit¹, Satendra K. Nirala² and Monika Bhaduria^{1,*}

¹Toxicology and Pharmacology Laboratory, Department of Zoology Guru Ghasidas University, Bilaspur, 495009 (C.G.) India; ²Laboratory of Natural Products, Department of Rural Technology and Social Development Guru Ghasidas University, Bilaspur 495009 (C.G.) India

Abstract: Background: Secondary metabolites of plants, the polyphenols, play a vital role in protection from many health problems in human beings. Structurally favored phytochemicals may be studied to protect multiorgan injury. At pharmacological doses, gallic acid is nontoxic to mammals and is generally absorbed in the intestine.

Aims: In this present study, gallic acid was evaluated for its protective efficacy against Lipo Polysaccharide (LPS) and d-Galactosamine (D-GalN) induced multiorgan injury, i.e., liver, kidney and brain.

Methods: Three different doses of gallic acid (5, 10 and 20 mg/kg p.o.) were administered to the experimental animals for 6 consecutive days, followed by exposure to LPS (50 µg/kg I.P.) and D-GalN (300 mg/kg I.P.) on the 6th day.

Results: Exposure to LPS and D-GalN resulted in increased oxidative stress and proinflammatory cytokines. Altered hematology and serology due to LPS and D-GalN were restored towards control by gallic acid. Declined antioxidants such as reduced glutathione, superoxide dismutase and catalase due to injurious effects of LPS and D-GalN were rejuvenated by gallic acid.

Discussion: Exposure to LPS and D-GalN severely increased lipid peroxidation, CYP2E1 activity and tissue lipids while lowered protein content. Gallic acid restored all these parameters towards control in dose dependent manner and 20 mg/kg dose provided the best protection. Histological study showed improved histoarchitecture of liver, kidney and brain that supported biochemical endpoints.

Conclusion: Gallic acid minimized oxidative stress and provided best protection at 20 mg/kg dose against LPS and D-GalN induced multi organ acute injury.

ARTICLE HISTORY

Received: December 16, 2019
Revised: March 30, 2020
Accepted: March 31, 2020

DOI:
10.2174/1389201021666200615165732



CrossMark

Keywords: Gallic acid, biochemical, lipopolysaccharide, D-galactosamine, antioxidant, multiorgan injury.

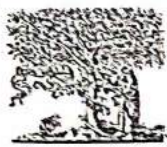
1. INTRODUCTION

Acute Liver Failure (ALF) or fulminant hepatic failure is characterized by nephritis, encephalopathy and coagulopathy in patients with previously normal liver physiology [1], which consequently produce multi organ injury. Fulminant hepatic failure may be induced by bacteria, viral hepatitis, hepatitis B virus, alcohol and other hepatotoxic agents [2] like acetaminophen, which accounts for more than 50% of all cases in some western societies [3, 4]. The ALF is a result of a massive necrosis in liver. Due to high mortality in the patients, ALF has no better treatment except liver transplantation [5, 6]. About 90% of acute hepatitis happens due to viruses, which can also produce primary liver cancer and

currently there are no effective therapeutic strategies against acute liver injury [7]. A combination of Lipopolysaccharide (LPS) and D-Galactosamine (D-GalN) causes acute liver injury in animals, which closely resembles the immunometabolic dysfunctions as seen in the clinical syndrome [8] and correlates with human acute liver failure complicated by endotoxemia or septicemia [9]. The LPS along with D-GalN enhances oxidative stress due to increased formation of reactive oxygen species, which may cause injury in various organs of the body, including liver, kidney and brain [10].

The medicinal plants attracted the attention of researchers for potential treatment of various ailments with highly safer use and minimal side effects [11]. Certain natural compounds or their derivatives were found to have the ability to protect cells from oxidative stress and related diseases [12]. Polyphenols have good oral bioavailability and have been used for disease prevention for decades [13]. Gallic acid, 3,4,5-trihydroxy benzoic acid is a colorless crystalline organic

*Address correspondence to this author at the Toxicology and Pharmacology Laboratory Department of Zoology, Guru Ghasidas University Bilaspur (C.G.) India; Tel: +91-9407567647; E-mail: monikabhaduria@rediffmail.com



ELSEVIER

Contents lists available at ScienceDirect

Sustainable Chemistry and Pharmacy

journal homepage: <http://www.elsevier.com/locate/scp>

Microwave hydrodiffusion and gravity model with a blend of high and low power microwave firing for improved yield of phenolics and flavonoids from oyster mushroom

Kavi Bhushan Singh Chouhan^a, Roshni Tandey^a, Kamal Kumar Sen^b, Rajendra Mehta^b, Vivekananda Mandal^{a,c}^a Institute of Pharmacy, Guru Ghasidas Central University, Bilaspur, CG, 495009, India^b Department of Rural Technology, Guru Ghasidas Central University, Bilaspur, 495009, India

IF-3.294

ARTICLE INFO

Keywords
Oyster mushroom
Microwave hydrodiffusion and gravity
Microwave assisted extraction
Phenolics
Antioxidants

ABSTRACT

Oyster mushrooms are a globally important entity both commercially and therapeutically with immense food value. The work describes a successful application of a unique method of microwave hydrodiffusion and gravity combining optimal mix of high and low power microwave firing for the extraction of phenolics and flavonoids principles from oyster mushroom for the first time. Initial firing of microwaves at higher power level (510 W for 2 min and 340 W for 2 min) was applied followed by sustained microwave firing at 170 W till the completion of the extraction as indicated by physical oozing out of the aqueous extract. Such a combination of microwave firing in microwave hydrodiffusion and gravity (MHG) method (26 min) successfully produced two times more yield of extract with $57.7\% \pm 2.8$ and $82.3\% \pm 3.7$ more content of phenolics and flavonoids, respectively, when compared to 5 h Soxhlet extraction and that too with a better reproducibility. Mapping of individual phenolics/flavonoids were carried out and SEM images were studied for better understanding of the operational aspects. The research aims towards canvassing green technologies as in the near future only those technologies shall survive which are in tandem with environment.

1. Introduction

Oyster mushroom is one of the most popular edible mushrooms with high nutritional value and also can be easily cultivated from various agro-industrial residues with minimal effort. Recently these mushrooms have gained added advantage owing to their rapid use as functional foods due to their potential beneficial effects on human health which makes them a high priority commercial entity for the food industry (Corrêa et al., 2016). Apart from their nutritional richness in terms of presence of various phenolics/flavonoids and other antioxidants; oyster mushrooms also possess potent antinociceptive, antitumor, antioxidant, and immunological activities. A comprehensive review on the health promoting aspects of mushrooms and on the nutritional and therapeutic benefits of oyster mushroom in particular have been vividly crafted and published by Ma et al. and Correa et al. respectively (Corrêa et al., 2016; Ma et al., 2018).

Having said and understood about the therapeutic and commercial significance of oyster mushroom, it becomes need of the hour to adopt to

green and sustainable extraction methods for large scale economical extraction of nutraceutical principles so that the growing demands of food industries can be met. In this regard, industries dealing with natural products as the source of raw material need to apply various solvents to accomplish the task of extract preparation (Kala et al., 2016). In such cases, contamination of finished product with trace residue of solvent used in the extraction process becomes a serious issue (Chemat et al., 2020). When dealing with plant derived extracts, at the production level it must be fully ensured that the finished product (dry extract) should be free from contamination particularly traces of any organic solvent which was used during the extraction process. Henceforth, an approach of microwave hydrodiffusion and gravity (MHG) was adopted which does not make use of any solvent and relies on extraction of contents from the ruptured cells and glands due to microwave heating. MHG basically is an innovative "upside down" extension of microwave assisted extraction system (Li et al., 2013). It combines microwave heating and earth's gravity at atmospheric pressure. This technique is best applicable for plant matrix which contains moisture (Bittar et al., 2013; Ciriminna

* Correspondence author.

E-mail addresses: pharmafriend@rediffmail.com, v.mandal@ggu.ac.in (V. Mandal).<https://doi.org/10.1016/j.scp.2020.100311>Received 23 May 2020; Received in revised form 27 July 2020; Accepted 1 August 2020
2352-5541/© 2020 Elsevier B.V. All rights reserved.

Hydro-ethanolic extract of *Gloriosa superba* protecting acute splenic injury by modulating endogenous antioxidants pool in rats

Piyush Shukla, Naresh Kumar Sahu, Raj Kumar, Monika Bhadauria¹ and *Satendra Kumar Nirala

Department of Rural Technology and Social Development

¹Department of Zoology

Guru Ghasidas University,
Koni - BILASPUR-495009 (C.G.) INDIA

*Corresponding Author

satendrak75@rediffmail.com

Received : 12.08.2020; Accepted : 22.09.2020

ABSTRACT

Present investigation was designed to evaluate hydro-ethanolic extract of roots of *Gloriosa superba* against combined exposure to alcohol and LPS induced splenic injury in rats. Thirty female albino rats were divided into five groups having six animals in each. Splenic injury was induced by combined exposure to alcohol and lipopolysaccharide (LPS) and 50, 100 and 200 mg/kg dose of extract of *Gloriosa superba* was evaluated for its efficacy. Combined exposure to alcohol and LPS significantly decreased total proteins but significantly increased triglycerides, cholesterol and lipid peroxidation process in spleen tissues, indicating metabolic perturbation and peroxidative cellular damage. Co-exposure to alcohol and LPS significantly decreased activity of superoxide dismutase, catalase, glutathione reductase, glutathione peroxidase, glucose-6-phosphate dehydrogenase as well as diminished reduced glutathione level in splenic tissue samples indicating exhaustion of endogenous antioxidant pool and development of oxidative stress. Conjoint administration of extract of *Gloriosa superba* at all the three doses offered significant protection in studied endpoints and maintained their values towards their respective control in dose dependent manner. It may be concluded that hydro-ethanolic extract of *Gloriosa superba* has potential to protect splenic injury by modulating endogenous antioxidants pool due to antioxidant phytochemicals present in extract.

Figures : 02

References : 27

Table : 00

KEY WORDS : Alcohol, Antioxidant pool, *Gloriosa superba*, Lipopolysaccharide (LPS), Oxidative stress.

Introduction

Co-exposure to alcohol and bacterial endotoxin, lipopolysaccharides (LPS) are very common in rural areas where societal alcoholism and unhygienic conditions prevail, which induce oxidative stress in population. Oxidative stress is defined as a status of an imbalance between cellular anti-oxidative capacity and reactive oxygen species (ROS) formation caused by deregulation of antioxidant system¹⁰. Alcohol-induced oxidative stress is linked to metabolism of ethanol involving both microsomal and mitochondrial systems. Ethanol metabolism is directly involved in the production of reactive oxygen species (ROS) and reactive nitrogen species (RNS)⁹. The LPS are amphiphilic molecules that are localized in the outer leaflet of the outer membranes of Gram-negative bacteria⁹. Combined exposure to alcohol and LPS produces multiple organ damage, including splenic injury, which consequently lead to immune

dysfunction, hemopoietic disorder and many other health issues. Despite of important functions of several vital organs in an organism, the spleen plays multiple supporting role in body. It acts as a filter for blood as part of the immune system. Old red blood cells are recycled in the spleen and platelets and white blood cells are stored there. The spleen also helps to fight certain kinds of bacteria that cause pneumonia and meningitis. Bacterial infection triggers a response that involves production of inflammatory mediators such as cytokines²⁶. In contrast to the general decrease in hypothalamic cytokine expression in LPS-pre-treated rats, an increase in most cytokines was observed in the spleen²⁰.

Nature has been a source of medicinal agents for thousands of years. Herbal medicine represents one of the most important fields of traditional medicine all over the world¹¹. The *Gloriosa superba* Linn. has important place in the Indian medicine, which cures many

ACKNOWLEDGEMENT : Financial assistance from UGC-Start up Research Project (No: F.20-1/2012 (BSR)/20-11 (3)/2012(BSR); Dated 30-10-2012) is gratefully acknowledged.



Naringenin mitigates antituberculosis drugs induced hepatic and renal injury in rats

Nisha Sahu^a, Gita Mishra^a, Hemeshwer Kumar Chandra^a, Satendra Kumar Nirala^b,
Monika Bhadauria^{a,*}

^a Toxicology and Pharmacology Laboratory, Department of Zoology, Guru Ghasidas University, Bilaspur, 495009 (CG), India

^b Laboratory of Natural Products, Department of Rural Technology and Social Development, Guru Ghasidas University, Bilaspur, 495009 (CG), India



ARTICLE INFO

Article history:

Received 16 July 2018

Received in revised form

2 January 2019

Accepted 3 January 2019

Available online 12 January 2019

Keywords:

Antituberculosis drugs

Naringenin

Hepato-renal injury

Histopathology

Antioxidant activity

Oxidative stress

ABSTRACT

Tuberculosis is one of the deadly diseases, which can be well treated by antituberculosis drugs (ATDs) i.e. isoniazid, rifampicin, pyrazinamide and ethambutol. These drugs also lead to severe hepatic and renal injury. The present study was designed to investigate efficacy of naringenin against ATDs induced hepato-renal injury. Rats were administered with ATDs for 8 weeks (3 day/week) followed by naringenin at three different doses (10, 20 and 40 mg/kg) conjointly for 8 weeks (3 days/week) orally. Silymarin (50 mg/kg) was used as positive control in the study. Hepatic and renal injury was measured by increased level of serological parameters such as aspartate aminotransferase, alanine aminotransferase, alkaline phosphatase, bilirubin, urea, uric acid and creatinine. The toxic effect of ATDs was also indicated by significant increase in lipid peroxidation along with decline in GSH, catalase and superoxide dismutase activity in liver and kidney tissues. Treatment with naringenin encountered ATDs induced injury as evident by significant reversal of biochemical indices towards their respective control in a dose dependent manner. Histopathological observations also supported biochemical findings. Assessment of TNF- α indicated therapeutic efficacy of naringenin at molecular level. Thus, results of this study clearly showed that naringenin possess protective role against ATDs induced hepato-renal injury and to take naringenin supplementation as food may be worthwhile to reduce ATDs induced hepato-renal injury.

© 2019 Center for Food and Biomolecules, National Taiwan University. Production and hosting by Elsevier Taiwan LLC. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

1. Introduction

Globally, tuberculosis has been declared as second leading cause of death from an infectious disease after the HIV.¹ A multi-therapy of first-line anti tuberculosis drugs (ATDs) isoniazid, pyrazinamide, rifampicin, and ethambutol is used for treatment of tuberculosis for 2 months followed by a combination of isoniazid/rifampicin for 8 months. Administration of isoniazid alone for 9 months causes 1.6% of hepatic damage however that increases to 2.6% in combination with rifampicin.² The ATDs cause various minor and major adverse reactions in body including skin reactions, gastrointestinal disorder, neurological disorder, vertigo and the most fatal one is

hepatotoxicity.^{3,4} The ATDs are metabolized in liver and produce various toxic reactive metabolites such as hydrazine, which form highly reactive oxygen species that act as stimulators of lipid peroxidation and disturbance in antioxidant defense system resulting in cell death.⁵ The ATDs not only cause aberrations in liver as it is the main detoxifying site but also severely effect kidney because of its close association with liver as an excretory organ. Thus, kidneys get exposed to harmful effect of intermediate or finished toxic metabolites and get injured in many cases.

The efficacy and safety of herbal and natural product have turned the major pharmaceutical population towards medicinal research.⁶ Naringenin is a plant bioflavonoid, mainly found in grape fruits, tomatoes and citrus fruits and is considered to have beneficial effects on human health.⁷ Many studies have explored the role of naringenin in improving health. Naringenin exerts various biological properties that includes anti-inflammatory, antioxidant, immunomodulatory, hepatoprotective, nephroprotective and neuroprotective.^{8–11} Management of tuberculosis treatment without

* Corresponding author. Toxicology and Pharmacology Laboratory, Department of Zoology, Guru Ghasidas (Central) University, Bilaspur 495009 (CG), India, E-mail address: bhadauria_monika@rediffmail.com (M. Bhadauria).

Peer review under responsibility of The Center for Food and Biomolecules, National Taiwan University.

<https://doi.org/10.1016/j.jtcm.2019.01.001>

2225-4110/© 2019 Center for Food and Biomolecules, National Taiwan University. Production and hosting by Elsevier Taiwan LLC. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

Protective role of rutin against combined exposure to lipopolysaccharide and D-galactosamine-induced dysfunctions in liver, kidney, and brain: Hematological, biochemical, and histological evidences

Samrat Rakshit¹ | Piyush Shukla² | Anjani Verma¹ | Satendra Kumar Nirala² |
Monika Bhadauria¹

¹Department of Zoology, Toxicology and Pharmacology Laborator. Guru Ghasidas University, Bilaspur, India

²Department of Rural Technology and Social Development, Laboratory of Natural Products, Guru Ghasidas University, Bilaspur, India

Correspondence

Monika Bhadauria, Department of Zoology, Toxicology and Pharmacology Laboratory, Bilaspur, India.
Email: monikabhadauria@rediffmail.com

Abstract

Protective efficacy of rutin over liver, kidney, and brain dysfunctions was evaluated in this investigation. Rutin (5, 10, and 20 mg/kg) was administered continuously for 6 days followed by single dose of D-galactosamine (300 mg/kg I.P.) and lipopolysaccharide (50 µg/kg I.P.) on the 6th day. Hematological, serological, biochemical, and histological aspects were considered for this study. One-way ANOVA ($p \leq .05$) followed by Tukey's HSD *post hoc* test determined the statistical significance. Serum AST, ALT, ALP, urea, uric acid, and creatinine were increased significantly, whereas albumin and glucose were significantly decreased after combined exposure to LPS and D-GalN. Glutathione level and activity of SOD and catalase were decreased, whereas lipid peroxidation, triglycerides, and cholesterol were increased in tissue samples due to LPS- and D-GalN-induced toxicity. Prophylactic treatment of rutin maintained studied variables toward control claiming the protective role of rutin.

Practical application: Rutin is plenteous in a variety of commonly ingested foods such as onion, wine, grape, citrus fruits, tea, and buckwheat. Rutin supplement is recommended for the treatment of various diseases such as varicose veins, internal bleeding, or hemorrhoids. Rutin is better than well-known antithrombic agent, Juniferdin, or Bacitracin. In the present study, rutin showed protective effects against LPS- and D-GalN-induced multiorgan dysfunctions due to its anti-inflammatory and antioxidant properties. Therefore, rutin may be developed and practiced as a food supplement to cope with acute organ dysfunctions caused by inflammatory and oxidative damage.

KEYWORDS

Antioxidant, Biochemical, D-galactosamine, lipopolysaccharide, oxidative stress, rutin

1 | INTRODUCTION

In the era of industrial and social advancement, human being remain in the vicinity of a plethora of toxic substances, including drugs, which not only affect one organ at a time but also impose

threat to other organ simultaneously. The two hit effect of combined exposure to lipopolysaccharide (LPS) and D-galactosamine (D-GalN) is well established model to study multiorgan damage and dysfunctioning. Combination of LPS and D-GalN induces hepatic injury is similar to clinical viral hepatitis; hence, this model



Pharmacognostic Characterization for Taxonomic Identification of *Bacopa monnieri* (L.) Wettst. for Quality Control

Naureen Shaba Khan^{1*}, Bhaskar Chaurasia² and Ashwini Kumar Dixit³

¹Assistant Professor, Department of Life Science, Dr. C.V. Raman University, Kota, Bilaspur, Chhattisgarh 495113, India.

²Assistant Professor, Department of Rural Technology & Social Development, Guru Ghasidas Vishwavidyalaya, Chhattisgarh, India.

³Associate Professor, Department of Botany, Guru Ghasidas Vishwavidyalaya, Koni; Bilaspur, Chhattisgarh 495009, India.

Abstract: Pharmacognostic studies played a crucial role in checking the possibilities and threat of adulteration among natural drugs by macroscopic and microscopic examination of plant materials. Organoleptic characters, macroscopic and microscopic studies, physicochemical and fluorescence analysis of *Bacopa monnieri* (L.) Wettst. has been carried out to determine the anatomical and pharmacognostic portrayal. The organoleptic studies revealed that *B. monnieri* is green in colour, bears a characteristic smell, leaves have smooth soft texture and bitter in taste. Stomata of anomocytic type and stomatal index was 17.77 and 18.18 in upper and lower epidermis respectively. Glandular trichome was present on the leaf surface and petiole. Study of powder microscopic analysis and fluorescence characteristics reveals round and oval starch granules, sclerenchymatous cells, lignified fibres, vascular bundles, spherical oil globules, raphides, druses, thin fibres and compounds oil droplets, xylem vessels with spiral and reticulates thickenings in the powder of different part of the plant. Observations revealed that the study has a great potential in unraveling the identification keys of plants species by highlighting various pharmacognostic aids.

Keywords: *Bacopa monnieri*; Bramhi; Pharmacognostic studies; Medicinal plants.

*Corresponding Author

Naureen Shaba Khan, Assistant Professor, Department of Life Science, Dr. C.V. Raman University, Kota, Bilaspur, Chhattisgarh 495113, India.



Received On 23 March 2020

Revised On 19 May 2020

Accepted On 08 June 2020

Published On 04 January 2021

Funding This research work was funded by UGC grant, New Delhi via financial assistance Ref no. FI-17.1/2013-14/MANF-2013-14-MUS-CH

Citation Naureen Shaba Khan, Bhaskar Chaurasia and Ashwini Kumar Dixit, Pharmacognostic Characterization for Taxonomic Identification of *Bacopa monnieri* (L.) Wettst. for Quality Control.(2021).Int. J. Life Sci. Pharma Res.11(1), L54-62
<http://dx.doi.org/10.22376/ijpbs/lpr.2021.11.1.L54-62>

This article is under the CC BY-NC-ND Licence (<https://creativecommons.org/licenses/by-nc-nd/4.0>)
Copyright © International Journal of Life Science and Pharma Research, available at www.ijlpr.com





Real-Time Identification of Fuzzy PID-Controlled Maglev System using TLBO-Based Functional Link Artificial Neural Network

Amit Kumar Sahoo¹ · Sudhansu Kumar Mishra² · Babita Majhi³ · Ganapati Panda⁴ · Suresh Chandra Satapathy⁵ 

Received: 30 May 2020 / Accepted: 30 December 2020
© King Fahd University of Petroleum & Minerals 2021

Abstract

In this paper, the teaching–learning-based optimization-based functional link artificial neural network (FLANN) has been proposed for the real-time identification of Maglev system. This proposed approach has been compared with some of the other state-of-the-art approaches, such as multilayer perceptron–backpropagation, FLANN least mean square, FLANN particle swarm optimization and FLANN black widow optimization. Further, the real-time Maglev system and the identified model are controlled by the Fuzzy PID controller in a closed loop system with proper choice of the controller parameters. The efficacy of the identified model is investigated by comparing the response of both the real-time and identified Fuzzy PID-controlled Maglev system. To validate the dominance of the proposed model, three nonparametric statistical tests, i.e., the sign test, Wilcoxon signed-rank test and Friedman test, are also performed.

Keywords System identification · Maglev system · FLANN · TLBO · Fuzzy PID

1 Introduction

In the recent past, many articles have been published on the identification of a complex system, owing to its widespread use in various areas. System identification means the estimation of the parameters of a plant or matching the output

responses of the model with that of the physical system. System identification is intended to find the deep understanding of the cause–effect relationships [1–4]. The nature of the system is categorized by different characteristics, such as its electrical, physical and chemical properties. However, it is very difficult to understand and model such characteristics of the plant. Thus, identification is a big challenge in several fields like control engineering [5, 6], power system engineering [7], renewable [8], etc. Accurate and quick identification is a difficult task for real-world plants which is mainly due to its nonlinear and dynamic nature. Many researchers have applied various forms of the artificial neural network (ANN) like multilayer perceptron (MLP) [9], functional link artificial neural network (FLANN) [10, 11], radial basis function (RBF) [12, 13], etc., for the identification purpose. By using multilayer perceptron (MLP) networks, Narendra and Parthasarathy have reported various identification techniques for a low complexity dynamic system [14]. However, the MLP network has multiple layers, which make it computationally expensive for the identification of any complex system. The FLANN model which is introduced by Pao et al. [15] is a single layer neural network without any hidden layer. The FLANN input is functionally expanded with different expansion techniques like power series, trigonometric, Chebyshev expansion, etc. This model is having lower computational complexity with a fast rate of convergence. The FLANN has

✉ Suresh Chandra Satapathy
suresh.satapathyfcs@kiit.ac.in

Amit Kumar Sahoo
amitkumar2687@gmail.com

Sudhansu Kumar Mishra
sudhansumishra@bitmesra.ac.in

Babita Majhi
babita.majhi@gmail.com

Ganapati Panda
ganapati.panda@gmail.com

¹ Department of EEE, Centurion University of Technology and Management, Bhubaneswar, Odisha, India

² Department of EEE, Birla Institute of Technology, Mesra, Ranchi, India

³ G. G. Vishwavidyalaya (A Central University), Bilaspur, CG 495009, India

⁴ C. V. Raman Global University, Bhubaneswar, India

⁵ KIIT University, Bhubaneswar, India



been used for pattern classification [16], prediction [17] and many other challenging tasks with faster convergence and lesser complexity compared to the MLP.

In the training phase of an ANN, all the weights are iteratively updated, and they reached the optimal value. The methods for updating the weights of a neural network could be based on the derivative or free from the derivative. Some of the examples of the derivative/gradient based are least mean squares (LMS) [6], backpropagation (BP) [2], recursive least squares (RLS) [18], etc. Similarly, examples of the second category include bio-inspired or evolutionary computing or computational intelligence-based approach. In most of the applications, the gradient-based approach provides inferior solutions due to the inherent limitations, such as trapping at local optimum points and incapability of finding derivatives of the discontinuous function.

To eliminate the above shortcomings, derivative-free algorithms, such as the genetic algorithm (GA) [19], particle swarm optimization (PSO) [20–22], and black widow optimization (BWO) [23], have been applied by different researchers to train the model. Kumar et al. [24] have introduced a metaheuristic-based socio-evolution and learning optimization algorithm (SELO) inspired by the social learning behavior of humans. The performance of the SELO is evaluated using 50 benchmark problems and compared with the other competitive algorithms. The results show the performance of the SELO is better than the others. Gholizadeh et al. [25] have introduced a metaheuristic algorithm, i.e., improved fireworks algorithm (IFWA) used for a discrete structural optimizations problems of steel trusses and frames. The optimization results demonstrate that the IFWA has highly competitive and superior over the standard FWA algorithm in terms of the convergence rate and statistical analysis. Gholizadeh et al. [26] have proposed a metaheuristic algorithm, center of mass optimization (CMO) to deal with performance-based discrete topology optimization (PBDTO) problem. PBDTO process is implemented for four multi-story steel braced frames by CMO. The authors have concluded that the CMO-based PBDTO formulation is an efficient technique for the seismic discrete topology optimization. Gholizadeh et al. [27] have proposed a new and efficient metaheuristic algorithm Newton metaheuristic algorithm (NMA) for optimization of steel moment frames. The NMA is a population-based framework which uses Newton gradient-based method. Here, the authors investigate the effectiveness of the proposed algorithm by considering two benchmark discrete trusses optimization problems. The performance of the proposed algorithm is analyzed on the basis of statistical parametric and nonparametric test and found to be superior over other competitive algorithms. Hayyolalam et al. [23] have proposed a novel black widow optimization algorithm (BWO), which is inspired by mating behavior of black widow spiders. The efficacy of the BWO

algorithm is determined by taking 51 different benchmark functions. From obtained results, it is confirmed that the BWO has better performance and superiority as compared to other algorithms. All these optimization techniques may be implemented to update the weights of the neural network and applied for identification of any system.

However, selecting the proper controlling parameters of these derivative-free bio-inspired algorithms is still a challenging task because of the presence of many controlling parameters. Due to these controlling parameters, the weight updation of neural network model is complex, computationally expensive and time consuming. Hence, there is a need to explore other bio-inspired algorithms with less number of controlling parameters. Rao et al. [28] recently came up with the TLBO optimization technique to circumvent the above shortcomings, which uses the teaching and learning methodology of the teacher and the student in a classroom. They highlighted the merits of TLBO that it does not depend on any controlling parameters, and only need the algorithm specific parameters, such as number of populations, iterations and stopping criteria. They have stressed on the fact that the TLBO eliminates the intricacy of the optimum selection and optimization of controlling parameters, which is usually necessary in other bio-inspired techniques. Naik et al. [29] have concluded that the performance of higher order neural networks is sensitive to weight initialization and relies on a kind of adopted learning algorithm. They have implemented TLBO for the training of ANN's, and applied it successfully for the classification problem. In this manuscript, we have implemented TLBO for optimizing weights of a variant of ANN, i.e., FLANN for identification of Maglev plant.

In this paper, MLP-BP, FLANN-LMS, FLANN-PSO, FLANN-TLBO and FLANN-BWO have been implemented for the identification of the Maglev system. The comparative analysis of performance among all these approaches is carried out by considering the mean squares error and the computational time. Here, a Fuzzy PID controller is also implemented to control the identified model, and then, the response is compared with that of the Fuzzy PID-controlled actual Maglev system.

The organization of the paper is as follows: Introduction and the recent work on identification are presented in Sect. 1. Section 2 presented and illustrated the construction and principle of the Maglev plant as shown in Fig. 1. Discussion of related work is presented in Sect. 3, and Sect. 4 highlighted the prerequisites of the research work. Section 5 deals with the proposed TLBO-based FLANN model for identification of the Maglev plant. In Sect. 6, design of controller based on the Fuzzy PID is discussed. In Sect. 7, the simulation study, validation and nonparametric statistical test of proposed model and the results of top-notch models are presented and compared. Section 8 presents the contribution of the manuscript, and the scope of future research work is outlined.

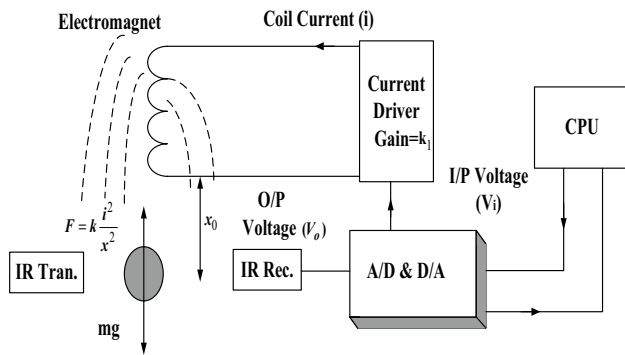


Fig. 1 Schematic diagram of the Maglev system

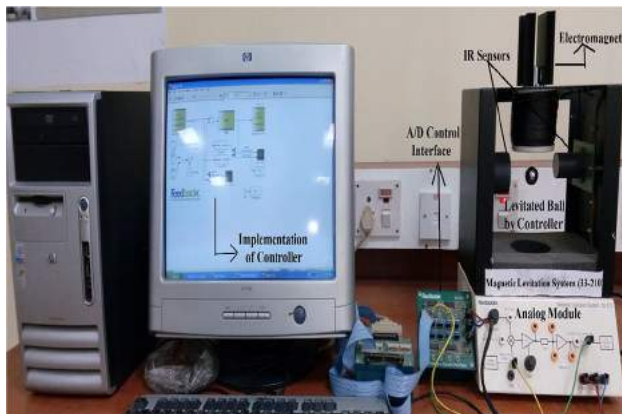


Fig. 2 The Maglev laboratory setup

2 The Magnetic Levitation Plant

The laboratory setup of the Maglev system from Feedback Instruments Ltd., Model No. 33-210 is shown in Fig. 2, and it has a wide range of applications like magnetically balanced bearings, vibration damping and transportation systems (i.e., very popularly known as Maglev trains) [30–33]. Basically, it works on the Maglev principle and has two parts: (i) the Maglev plant, and (ii) the digital computer where the controlling action takes place. The Maglev system comprises of different integrated components, like the electromagnet, ferromagnetic ball, IR sensor and a current driver circuit. A digital computer provides an immense platform for the effective design of various controllers, which can be implemented using MATLAB and Simulink for real-time applications. The whole setup accumulates both mechanical and electrical units with I/O interface systems.

The Maglev plant parameters are given in Table 1, and its transfer function is as follows [34–36]:

$$G_p(s) = \frac{\Delta V_o}{\Delta V_i} = \frac{-3518.85}{s^2 - 2180} \quad (1)$$

Table 1 The physical parameters of the Maglev system

Name of the parameter	Symbol	Value
Mass of steel ball	m	0.02 kg
Control voltage to current gain (constant)	k_1	1.05 A/V
Sensor gain (constant), offset	k_2, η	143.48 V/m, -2.8 V
Input voltage to the controller	V_i	+ 5 V and - 5 V
Output voltage of sensor	V_o	+ 1.25 V to - 3.75 V
Equilibrium position of steel ball	x_0	0.009 m
Current at equilibrium position	i_0	0.8 A
Gravitational constant	g	9.81m/s ²

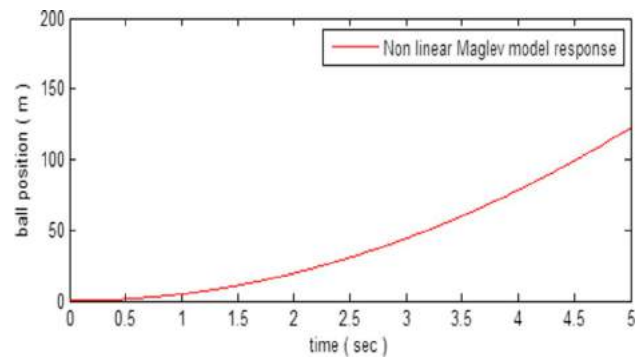


Fig. 3 Nonlinear response of Maglev system

where $G_p(s)$ represents the Maglev plant (Feedback Instruments Ltd., Model No. 33-210) transfer function, V_o is the output voltage of the sensor and V_i is the input voltage to the controller. From Eq. (1) and Fig. 3, it is found that the behavior of the Maglev system is highly nonlinear and unstable in nature. Therefore, it is challenging task to get the improved identified model of the Maglev plant.

3 Related Work

Artificial neural network (ANN) plays an important role in the identification of a nonlinear system [37, 38]. The neural network (NN) can performed nonlinear mapping between the input and output, as it has interconnection between the different layers. The neural network can be classified on the basis of its input, hidden and output layers. From the structural point of view, an ANN may be a single layer or multilayers. In a multilayer perceptron (MLP), there may be one or many hidden layers in between the input and output layers [39]. However, in a single layer structure, no hidden layer is present. Each neuron is connected from one layer to next layer of other neuron.

The learning of any neural network is a process where the weights are updated iteratively. These learning

processes may be of derivative based or derivative free. Some of the standard conventional gradient derivative-based approaches are LMS, RLS and BP algorithms. These have been applied by different researchers to train various neural networks and other adaptive models. Similarly, different derivative-free/evolutionary/bio-inspired learning algorithms, such as the GA, PSO, ant colony optimization (ACO), cat swarm optimization (CSO) and TLBO algorithms, are also used to train different neural network models.

Derivative-based algorithms are usually based on the gradient descent search algorithm and mathematically derived by utilizing the derivative of error. Least mean squares (LMS) are a stochastic gradient method or a simple derivative-based algorithm [40]. It is very popular, and widely used for its simple structure and ease of implementation to minimize the error. It is suitable for single layer ANN models for updating the weights. Backpropagation (BP) algorithm is derivative-based algorithm, which is suitable for multilayer ANN models [41]. The gradient-based optimization techniques fail to solve optimizing functions having discontinuities. These techniques may get trapped at local optimum points while solving functions having multiple optimal (maxima/minima) points. To overcome these bottlenecks of the traditional derivative-based approaches, different heuristic algorithms have been implemented by researchers. The PSO, which is based on the principle of the movement of a flock of birds that collectively search for food is a heuristic algorithm that has better convergence characteristics even for non-convex and discontinuous functions [20, 42]. This algorithm has a better exploration capability as the best among the swarm is followed by all the individuals along with their own best positions. The algorithm has provision for both local and global search techniques. The teaching–learning-based optimization (TLBO) has no control parameters. It undergoes a two phase search; the teacher phase performs a global search for better exploration, while the learner phase carries out for local search for better exploitation [43–45]. Also, this algorithm being dependent only on algorithm specific parameters, and without having controlling parameters is expected to have a better convergence characteristic is discussed in details in Sect. 4. The black widow optimization is a type of evolutionary-based optimization technique that imitates the strange mating behavior of the black widow spiders [23]. It is one of the latest techniques in the evolutionary-based optimization family. It delivers fast convergence speed, and avoids local optima problem. These techniques update the weights in three stages, i.e., procreate, cannibalism (sexual cannibalism and sibling cannibalism) and mutation.

4 Prerequisites

In this paper, a MLP and a special variant of ANN, i.e., FLANN is implemented for the identification of the Maglev system. FLANN is a type of single layer NN in which, the input data is allowed to pass through a functional expansion block, and hence, the input is functionally expanded with different expansion techniques. The power series expansion, trigonometric expansion and Chebyshev expansion are some of the mostly used expansion techniques. The Chebyshev functional expansion is found to be better for many engineering applications, and hence, it is considered for the expansion of FLANN inputs for the identification of Maglev system in this article. The Chebyshev expansion of input x_l , can be written as [41, 46, 47],

$$\begin{aligned}
 T_0(x_l) &= 1 \quad \text{for } l = 0 \\
 T_1(x_l) &= x_l \quad \text{for } l = 1 \\
 T_2(x_l) &= 2x_l^2 - 1 \quad \text{for } l = 2 \\
 T_{l+1}(x_l) &= 2x_l T_l(x_l) - T_{l-1}(x_l) \quad \text{for } l > 2
 \end{aligned}
 \tag{2}$$

The higher order polynomials are expanded as per usual practice. The output of the functional expansion block is multiplied with a set of weights. The basic structure of FLANN model that is trained by any adaptive algorithm is depicted in Fig. 4.

The lower computational complexity of the FLANN model due to its simple single layer structure, and simple learning algorithms, makes it computationally cheap and time efficient [48, 49]. The FLANN model holds the advantage of a single layer perceptron (SLP) network and an MLP network by evading their shortcomings. Here, the adaptive algorithm is the PSO and hence named the model as FLANN-PSO model. A set of input signals is given to the FLANN-PSO model, and the input of the FLANN model is functionally expanded nonlinearly by using the Chebyshev functional expansion technique. All the weights have been updated by using the PSO algorithm. Simultaneously,

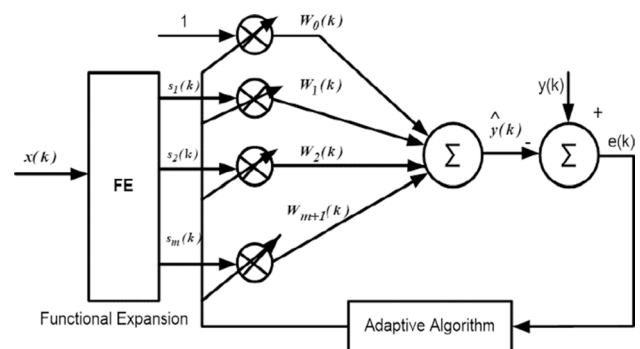


Fig. 4 Structure of the FLANN Model

a set of weights are initialized in between 0 and 1, and it is multiplied with the input signals. The output of each set of weights is compared with the same corresponding desired (output) signal and each set of weights is considered to be one particle. Hence, each set will produce one error signal.

The set of weights whose error is the minimum is considered to be the global particle. The other particles, i.e., other sets of weights, are local particles, and they update their velocity and position according to Eqs. (3) and (4):

$$V_i(d) = wV_i(d) + c_1 * rand * (P_i(d) - X_i(d)) + c_2 * rand * (P_g(d) - X_i(d)) \tag{3}$$

$$X_i(d + 1) = X_i(d) + V_i(d + 1) \tag{4}$$

where $V_i(d)$ and $X_i(d)$ represents the velocity and position of the i th particle, respectively, and $rand$ represents the random number, which is in between $[0,1]$. $P_g(d)$ and $P_i(d)$ are the position of g-best and p-best, respectively. w is the weighting factor, c_1 and c_2 are the constant whose values determine the effect of social and cognitive components.

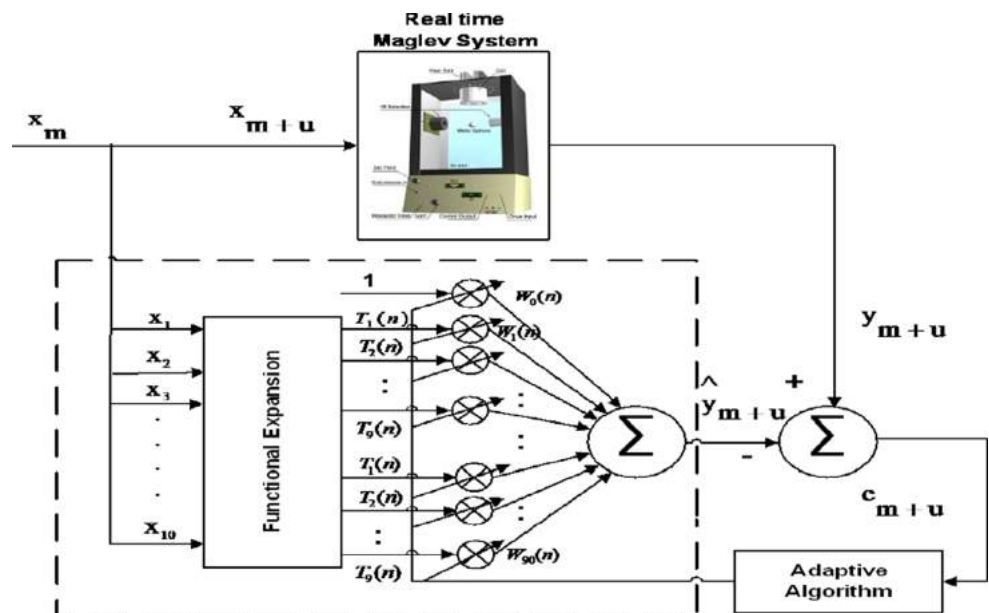
Once all the particles/set of weights are updated, again the error (i.e., objective function) will be calculated by using the new sets of weights. According to the minimum error, the respective weights will be saved so that it can be compared with the previous minimum error. If the current error is lesser than the previous one, then the current value is saved or the previous one. This process is repeated iteratively for a predefined number of times. After a certain epoch, the change of error is saturated, and then, the program is terminated. Finally, the optimum weight is reported. The FLANN network having this optimum weight is called as the trained network, and suitable for testing in the test data.

5 Proposed TLBO-Based FLANN Model

This article presents the metaheuristic TLBO technique based on the teaching and learning methodology, which helps to update the weights of FLANN [29]. The TLBO algorithm simulates a classroom like environment where the number of students is the population whose level of knowledge is considered as the possible solution set of the problem. Hence, the knowledge is defined by its objective function in the problem. The students in a classroom learn mainly through two processes; one through the teacher, and other by interacting between themselves. Thus, TLBO has two phases (a) the teacher phase and (b) the learner phase. In the ‘teacher phase,’ the learner group learns from the teacher, and in the ‘learner phase’ they learn by having discussions with one another. The most knowledgeable person in the classroom is considered as the teacher who shares his or her knowledge with the learners, and at every iteration, the best learner is considered as a teacher. Different designed variables of the optimization problem are analogous to the different subjects offered to the students (learners). The results (grade) of each learner are equivalent to the fitness of the problem. The teacher tries to enhance the knowledge of all the learners in accordance with his or her capability. The transfer of knowledge also depends on the capability of the students (learners).

A set of input signals having window size ‘ u ,’ i.e., $\{x_1, x_2, x_3, \dots, x_u\}$, is given to the proposed FLANN-TLBO model and again, and the input of the FLANN model is functionally expanded nonlinearly by using the Chebyshev functional expansion technique. Simultaneously, random sets of weights (equals to number of expanded inputs of

Fig. 5 Proposed FLANN model for identification of Maglev system



the FLANN) are initialized between 0 and 1. Each set is multiplied by the expanded input signals. Then, the output of the FLANN is compared with the desired signal. Hence, it results a set of error signal $\{e_1, e_2, e_3, \dots, e_u\}$.

Maglev plant input can be expanded by using Chebyshev expansion by the following mathematical form [46],

$$T_i(n) = \begin{bmatrix} T_1 \\ T_2 \\ T_3 \\ \vdots \\ T_k \end{bmatrix} = \begin{bmatrix} 1 \\ x_1 \\ 2x_1^2 - 1 \\ \vdots \\ 2x_k T_k(x_k) - T_{k-1} x_k \end{bmatrix} \text{ For } k > 2 \quad (5)$$

$$\hat{y}_{m+u} = \sum_{i=1}^{Q-1} T_i(n) w_i(n) \quad (6)$$

Here, x_m is the input, \hat{y}_{m+u} is the output of the FLANN model, $T_i(n)$ is the expanded input using Chebyshev expansion and $w_i(n)$ is the weight vector having Q no. of elements. Equation (6) shows the output of the proposed model as shown in Fig. 5. The weights set connected with the FLANN model is optimized by the TLBO algorithm to achieve desired response and the error is

$$e_{m+u} = y_{m+u} - \hat{y}_{m+u} \quad (7)$$

Fig. 6 Flowchart of FLANN-TLBO network

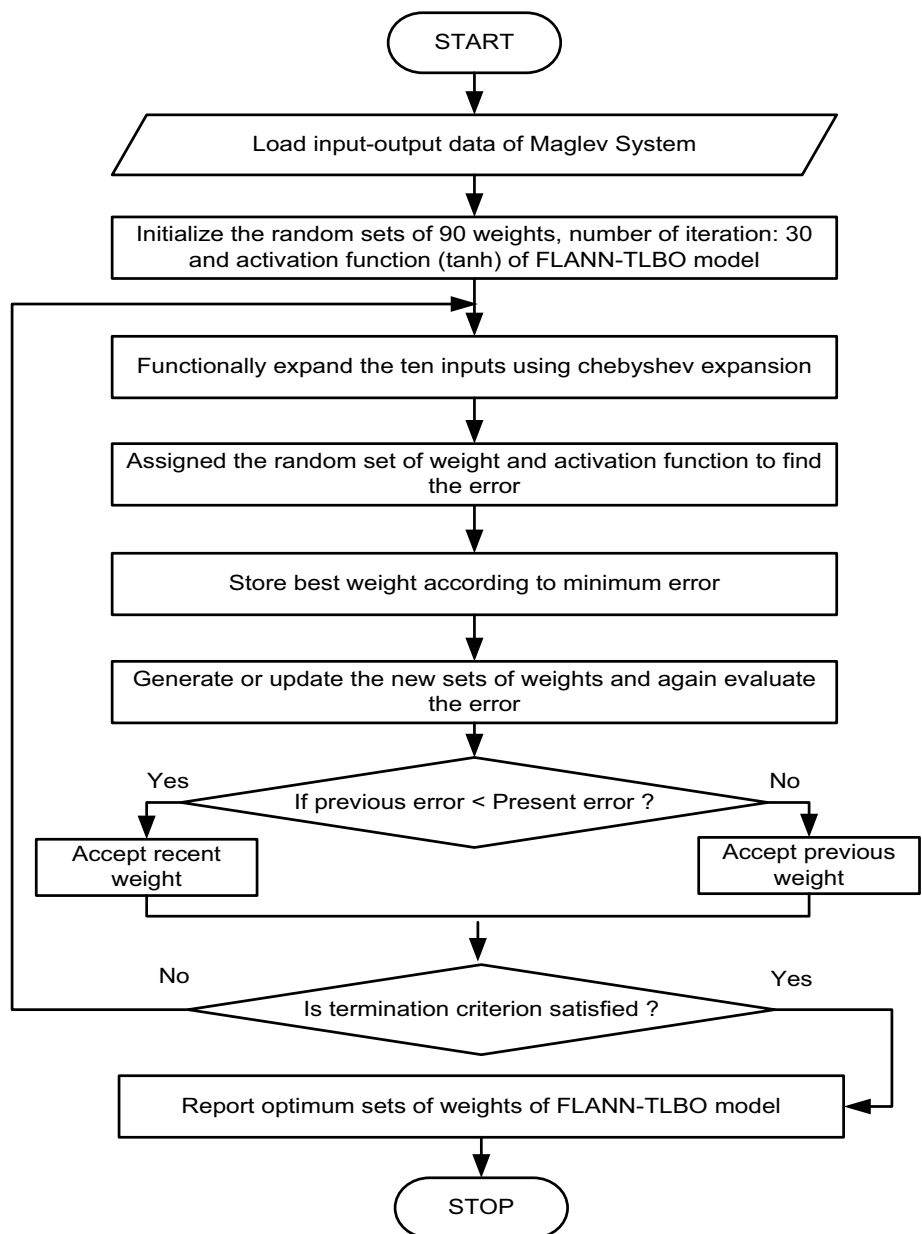


Fig. 7 Model of Fuzzy PID controller

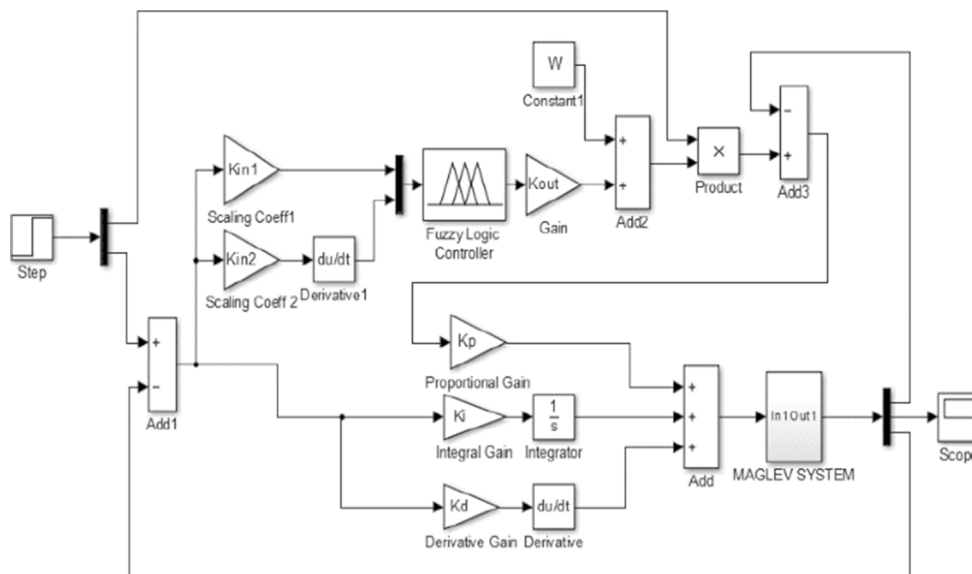


Table 2 Basic rule table for FIS

Row represents the error (e) and column represents the derivative of e					
	NB	NS	Z	PS	PB
NB	NVB	NB	NM	NS	Z
NS	NB	NM	NS	Z	PS
Z	NM	NS	Z	PS	PM
PS	NS	Z	PS	PM	PB
PB	Z	PS	PM	PB	PVB

Table 3 Linguistic variables of FIS

POSITIVE VERY BIG	PVB
POSITIVE BIG	PB
POSITIVE MEDIUM	PM
POSITIVE SMALL	PS
ZERO	Z
NEGATIVE SMALL	NS
NEGATIVE MEDIUM	NM
NEGATIVE BIG	NB
NEGATIVE VERY BIG	NVB

Here, the total input (m) = 1, 2, ..., $n-u$, input window size of FLANN (u) = 10 and n is the no. of expanded input.

5.1 Teaching phase

A teacher tries to enhance the performance of all the students in the class. Considering a class of n students (population size), m subjects (number of design variables) the mean

Fig. 8 Membership function of input variable

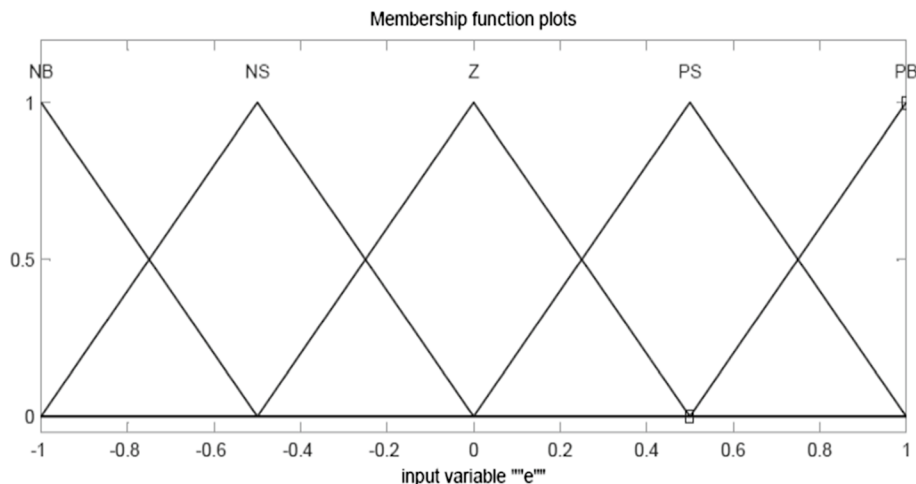
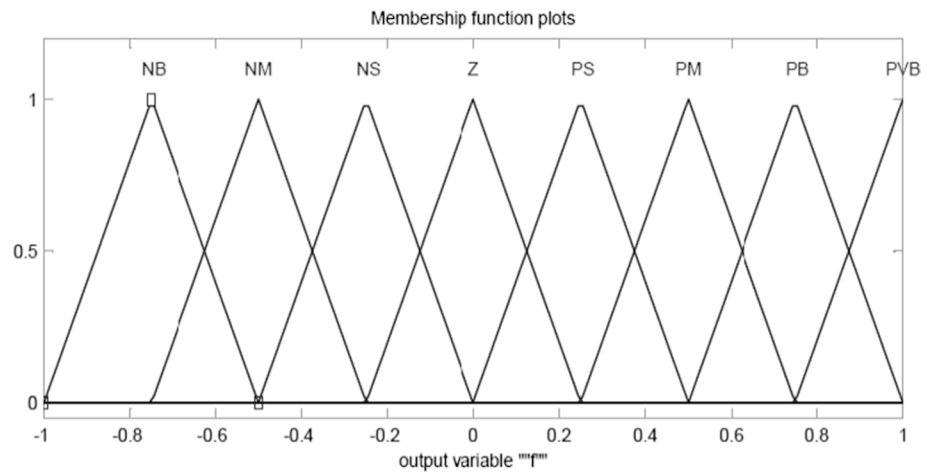


Fig. 9 Membership function for output variable



result of students in subject j can be denoted as M_{ji} for the i th iteration. Let the overall best result of all the subjects of the whole population obtained for the learner number be denoted by $kbest$. The teacher is the most knowledgeable person having the highest fitness value in the class. So, the teacher tries to improve the results of the students, for this a correction variable derived from the difference between the

mean results of the k th student and the teacher in subject j is defined as

$$DM_{j,k,i} = rd \times (W_{j,kbest,i} - T_f \times M_{ji}) \tag{8}$$

where DM correspond to the difference of mean, $W_{j,kbest,i}$ is the result of the teacher in j th subject, rd is a random number between 0 and 1. T_f is called as the teaching factor whose value is either 1 or 2. The teaching factor is defined randomly as

$$T_f = round[1 + rd(0, 1) * (2 - 1)] \tag{9}$$

The solutions are updated as

$$W_{j,k,i}^{new} = W_{j,k,i} + DM_{j,k,i} \tag{10}$$

where $W_{j,k,i}^{new}$ is the updated result value of the k th student in the j th subject at the i th iteration and $W_{j,k,i}$ is the existing result. However, the updated result will be accepted if it satisfies the boundary condition, else it has to be replaced by the limiting boundary value. Also, it should have a better fitness than that of the existing values; otherwise, it need not be replaced. This updated value will act as input to the learner phase (Fig. 5).

5.2 Learner phase

The individual learner enhanced his or her own knowledge by interacting with his/her classmates apart from learning from the teacher. It is a convention that a learner will learn from another learner if the other learner's knowledge is more than his or her. In this phase, two copies p and q are selected randomly such that $x'_{totalpi} \neq x'_{totalqi}$, i.e., the total results as updated in the teacher phase do not match. Then, their results are updated as:

$$x''_{jpi} = x'_{jpi} + r_i(x'_{jpi} - x'_{jq_i}); \text{ if } x'_{totalpi} < x'_{totalqi} \tag{11}$$

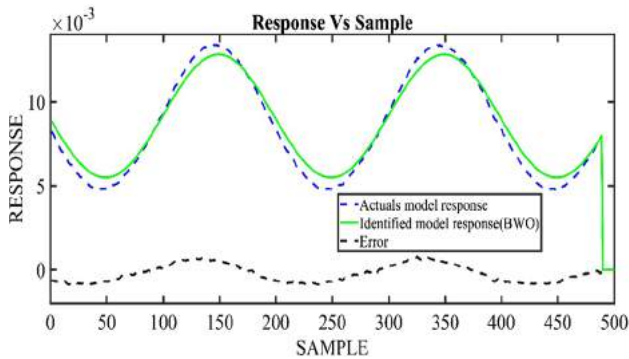


Fig. 10 Identified model response with FLANN-BWO

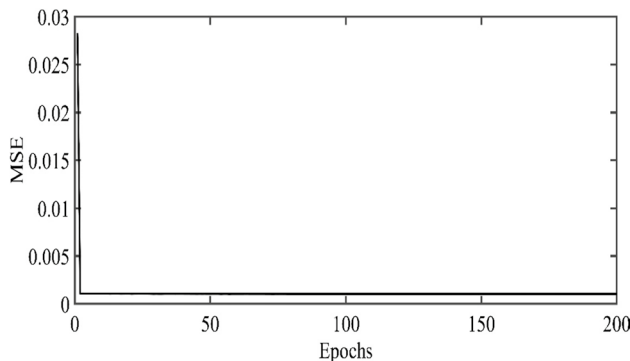


Fig. 11 MSE plot of FLANN-BWO

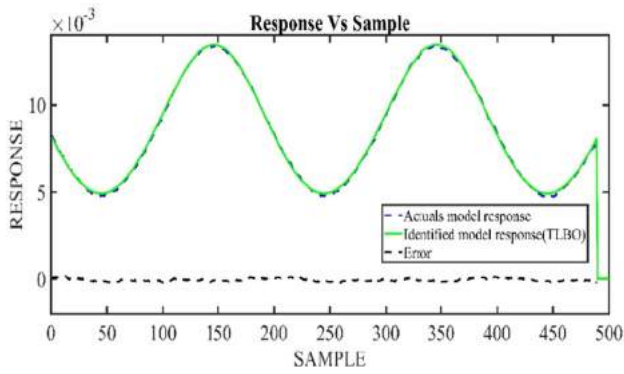


Fig. 12 Identified model response with FLANN-TLBO

$$x''_{jpi} = x'_{jpi} + r_i(x'_{jq_i} - x'_{jpi}); \text{ if } x'_{totalpi} > x'_{totalqi} \quad (12)$$

where x''_{jpi} is accepted if its fitness value is better than that of x'_{jpi} ; further, x''_{jpi} should satisfy the boundary condition. If the boundary condition is not satisfied, it should be replaced by x'_{jpi} .

The 1st set of updated weights and error values is stored for the forthcoming assessment. Again, the TLBO is applied to update the next set of weights and matched with the previous value of weights. The best set of weights, i.e., the set of weight having minimum error is considered to be the teacher, and the other sets are learners. The parameter of the proposed model undergoes the teaching and learning phase of TLBO to update the weights of the FLANN network. This process has been repeated until the error is less than the threshold value. The flowchart describes the detailed process of the TLBO-based FLANN model as shown in Fig. 6.

6 Design of the Fuzzy PID Controller

The universally accepted PID controller is an important tool for industrial control and automation, due to its reliability and adaptability [50]. It has the capability to handle

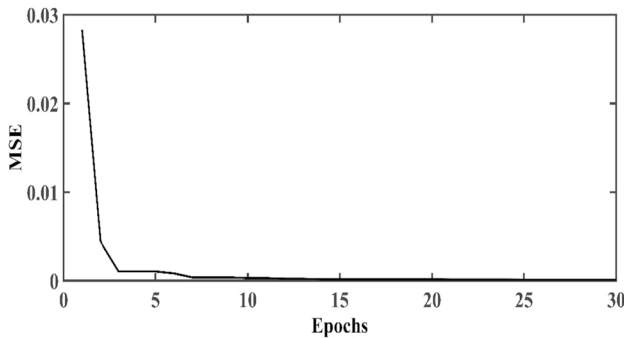


Fig. 13 MSE plot of FLANN-TLBO

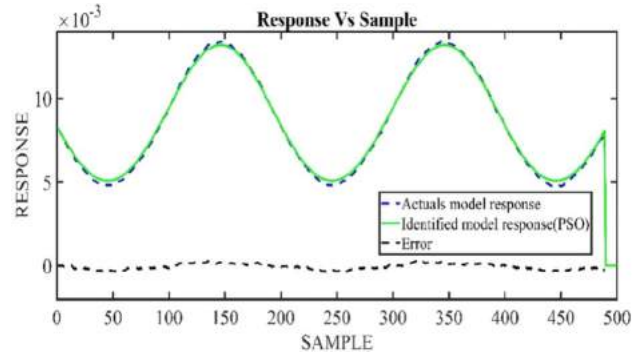


Fig. 14 Identified model response with FLANN-PSO

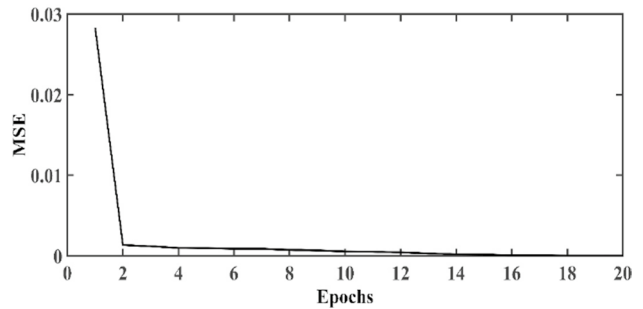


Fig. 15 MSE plot of FLANN-PSO

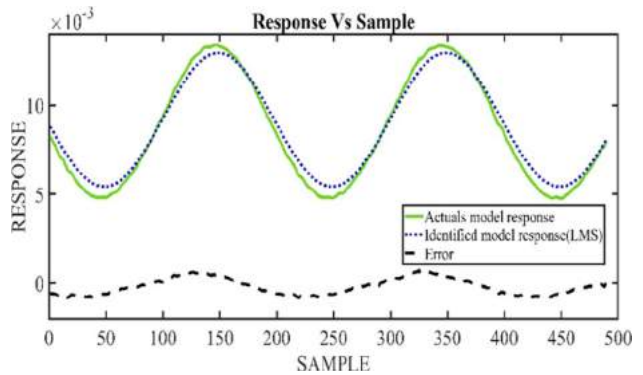


Fig. 16 Identified model response with FLANN-LMS

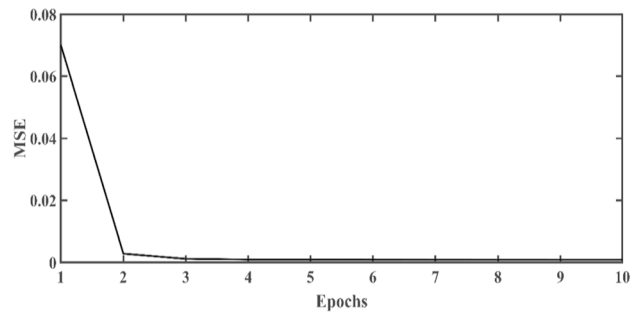


Fig. 17 MSE plot of FLANN-LMS

Table 4 Comparative results of identified model of Maglev system

Model	No. of iteration	Average MSE	CPU time (in s)	Big O Notation
FLANN-BWO	100	2.28E-07	382.422	$O(Ni * (Ps * Fe) * Nr)$
FLANN-TLBO	30	2.7498E-08	462.02	$O(Ni * (Ps * Fe) * Nr)$
FLANN-PSO	20	1.3945E-08	782.43	$O(Ni * (Ps * Fe) * Nr)$
FLANN-LMS	10	2.47E-07	4.15	$O(Ni * Nr)$
MLP-BP	20	1.1470E-07	8.96	$O(Ni * Nr)$

all the shortcomings of any other controller, and is found to be suitable for many of the industrial requirements. However, due to the high nonlinearity and uncertainty present in different systems, the performance of the PID gets degraded. To avoid these bottlenecks and enhance the capability of the PID controller, a fuzzy technique has been incorporated with the PID controller by researchers [51].

The control law associated with PID is as follows:

$$u(t) = k_p e(t) + k_i \int e(t)dt + k_d \frac{de(t)}{dt} \tag{13}$$

where k_p is the proportional gain, k_i is the integral gain, k_d is the derivative gains, $e(t)$ is the error signal and $u(t)$ is the control input.

The intervening system is fuzzified with two inputs, i.e., the system error (e) and derivative of error (\dot{e}), obtained from coefficients, K_{in1} and K_{in2} as shown in Fig. 7. These two values match the values between -1 and 1 . This leads to assign the membership function in a definite manner using the rule Table 2 and the linguistic variable Table 3.

For each input, five membership functions are chosen and assigned. However, for the output, nine triangular membership functions have been defined from -1 to 1 as shown in Figs. 8 and 9, and found from the coefficient K_{out} .

This Fuzzy PID controller has been utilized for validating our identified model. This controller has been implemented to the identified model, and in the real-time Maglev plant. The responses of the identified model and the actual model are compared to investigate the performance of the proposed.

7 Simulation Study

The algorithms were executed in the *Acer Aspire V* system, Window 10 OS, Intel® Core™ i5-3337U CPU @ 1.80 GHz processor, RAM of 8 GB and in a MATLAB environment. Five different neural network models, i.e., the MLP trained by BP, and FLANN networks trained by the LMS, PSO,

TLBO and BWO algorithms have been implemented for the comparative analysis.

7.1 Performance Analysis

All the possible functional expansions are implemented, and we found that the Chebyshev functional expansion model is found to be the most effective in our application. Hence, in our study, we have utilized the Chebyshev expansion in all the four FLANN models for reasonable comparison. The error signal which is the difference between the desired signal and the output of the FLANN network is considered to be the cost function. The following parameter have been considered for the identification of Maglev system using different algorithms.

Table 5 Comparative results of MSE of various optimization techniques for 20 independent test runs

Sl. no.	MSE				
	BWO	TLBO	PSO	LMS	MLP-BP
1	1.99E-07	6.14E-08	1.20E-07	2.71E-07	1.81E-07
2	2.59E-07	1.16E-08	5.83E-08	2.29E-07	1.12E-07
3	2.72E-07	7.93E-09	1.34E-07	2.23E-07	1.50E-08
4	2.25E-07	8.69E-09	4.04E-08	2.85E-07	1.43E-07
5	2.90E-07	2.01E-08	1.46E-07	2.64E-07	2.16E-07
6	2.28E-07	2.82E-08	1.56E-07	2.36E-07	2.56E-07
7	2.67E-07	2.47E-08	1.04E-07	3.02E-07	2.12E-07
8	2.02E-07	5.15E-08	1.33E-07	3.39E-07	3.30E-08
9	2.88E-07	5.66E-08	1.26E-07	1.72E-07	1.95E-07
10	2.44E-07	3.82E-08	1.56E-07	2.73E-07	1.99E-07
11	2.59E-07	4.77E-09	1.36E-07	2.24E-07	1.68E-07
12	2.62E-07	2.29E-07	1.19E-07	1.73E-07	2.27E-07
13	2.94E-07	7.82E-08	1.38E-07	2.45E-07	1.57E-07
14	2.38E-07	1.42E-07	6.61E-08	2.43E-07	1.89E-07
15	2.71E-07	3.40E-08	3.87E-08	3.09E-07	2.76E-07
16	2.75E-07	6.10E-09	8.38E-08	2.80E-07	3.30E-07
17	3.10E-07	5.90E-09	1.51E-08	1.72E-07	1.57E-07
18	2.83E-07	6.69E-08	1.02E-07	1.78E-07	3.80E-08
19	2.17E-07	6.36E-08	1.18E-07	2.28E-07	1.94E-07
20	1.71E-07	4.84E-09	1.26E-07	3.02E-07	2.86E-07

Table 6 The best sets of weight from FLANN-TLBO model ($W_1 - W_{90}$)

W_1	0.435779	W_{19}	0.827147	W_{37}	0.155891	W_{55}	0.264486	W_{73}	0.914621
W_2	0.682444	W_{20}	0.698345	W_{38}	0.112564	W_{56}	0.82409	W_{74}	0.79115
W_3	0.773512	W_{21}	0.320443	W_{39}	0.492218	W_{57}	0.25293	W_{75}	0.958493
W_4	-0.00529	W_{22}	0.953393	W_{40}	0.953885	W_{58}	0.938268	W_{76}	0.787987
W_5	0.781096	W_{23}	0.037445	W_{41}	0.334814	W_{59}	0.358586	W_{77}	0.16796
W_6	0.500079	W_{24}	0.306598	W_{42}	0.579984	W_{60}	0.204795	W_{78}	0.98138
W_7	-0.03474	W_{25}	0.249404	W_{43}	0.218814	W_{61}	0.258882	W_{79}	1.066246
W_8	0.146218	W_{26}	0.633354	W_{44}	0.883247	W_{62}	0.623441	W_{80}	0.81099
W_9	0.414602	W_{27}	0.663029	W_{45}	0.387094	W_{63}	0.480284	W_{81}	0.889997
W_{10}	0.875391	W_{28}	0.054693	W_{46}	0.637974	W_{64}	0.219913	W_{82}	0.825227
W_{11}	0.524488	W_{29}	0.357577	W_{47}	0.831112	W_{65}	0.699049	W_{83}	0.832609
W_{12}	0.787741	W_{30}	0.31339	W_{48}	1.022958	W_{66}	0.453451	W_{84}	0.156097
W_{13}	0.303452	W_{31}	0.514109	W_{49}	1.091364	W_{67}	0.417878	W_{85}	0.969134
W_{14}	0.710593	W_{32}	0.577152	W_{50}	0.679307	W_{68}	0.785315	W_{86}	0.955765
W_{15}	0.036209	W_{33}	0.622466	W_{51}	0.270734	W_{69}	0.153927	W_{87}	0.484031
W_{16}	0.281128	W_{34}	0.268449	W_{52}	0.281422	W_{70}	0.625255	W_{88}	0.798993
W_{17}	0.050205	W_{35}	0.672411	W_{53}	0.389655	W_{71}	0.621751	W_{89}	0.140657
W_{18}	0.100993	W_{36}	0.648091	W_{54}	0.851318	W_{72}	0.248435	W_{90}	0.420589

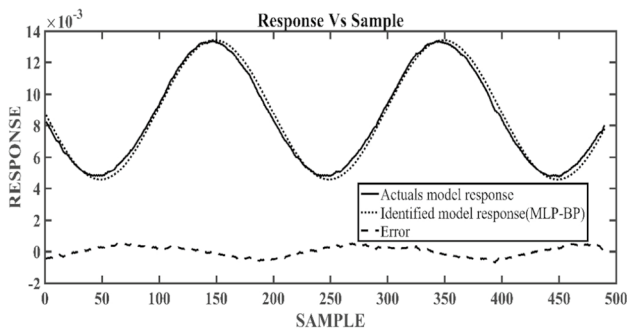


Fig. 18 Identified model response with MLP-BP

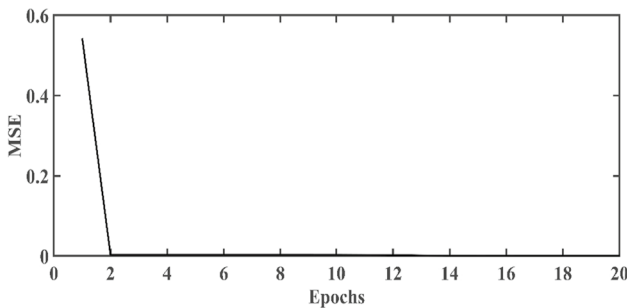


Fig. 19 MSE plot of MLP-BP

In FLANN-LMS: Learning rate (μ): 0.6, No. of iteration (N_i): 10, No. of weights: 90 and Activation function: *tanh*.

In FLANN-PSO: Learning rate (μ): 0.6, No. of iteration (N_i): 20, No. of feature (Fe): 20, cognitive parameter:

$c_1 = c_2 = 2$, Population size (Ps): 45, Inertia rate: 0.9, No. of weights: 90 and Activation function: *tanh*.

In FLANN-TLBO: Population size (Ps): 45, No. of iteration (N_i): 30, No. of feature (Fe): 20, No. of weights: 90 and Activation function: *tanh*.

In FLANN-BWO: Population size (Ps): 45, No. of iteration (N_i): 100, No. of feature (Fe): 20, Procreating rate (PP): 0.6, Cannibalism rate (CR): 0.675, Mutation rate (PM): 0.4, No. of weights: 90 and Activation function: *tanh*.

In MLP-BP: Learning rate (μ): 0.6, No. of iteration (N_i): 20, No. of layer: 3, Node: 5-3-1, No. of weights: 90 and Activation function: *tanh*.

To study the effectiveness of the proposed model, 5000 samples are taken. In Fig. 11, it is shown that the FLANN-BWO model has an average MSE of $2.28E-07$ after 100 iterations and the corresponding CPU time 382.422 s. The FLANN-TLBO model has an average MSE of $2.7498E-08$ after 30 iterations and CPU time 462.02 s as displayed in Fig. 13. But, the FLANN-PSO and MLP-BP models have average MSE of $1.3945E-08$ and $1.1470E-07$, respectively, after 20 iterations each and corresponding CPU time is 782.43 s and 8.96 s, respectively, as presented in Figs. 15 and 19. The gradient-based FLANN-LMS model shown in Fig. 17 has an average MSE of $2.47E-07$ after 10 iterations and CPU time of 4.15 s, which is the lowest among others. By taking the proposed model with different bio-inspired algorithms, the value of MSE has been reduced from $1.1470E-07$ to $2.7498E-08$, as listed in Table 4. After training of the proposed model, the best set of 90 weights, which represents the identified model of the Maglev system, is listed in Table 6. The fitting and MSE curves of all the models are shown in Figs. 10, 11, 12, 13, 14, 15, 16, 17, 18

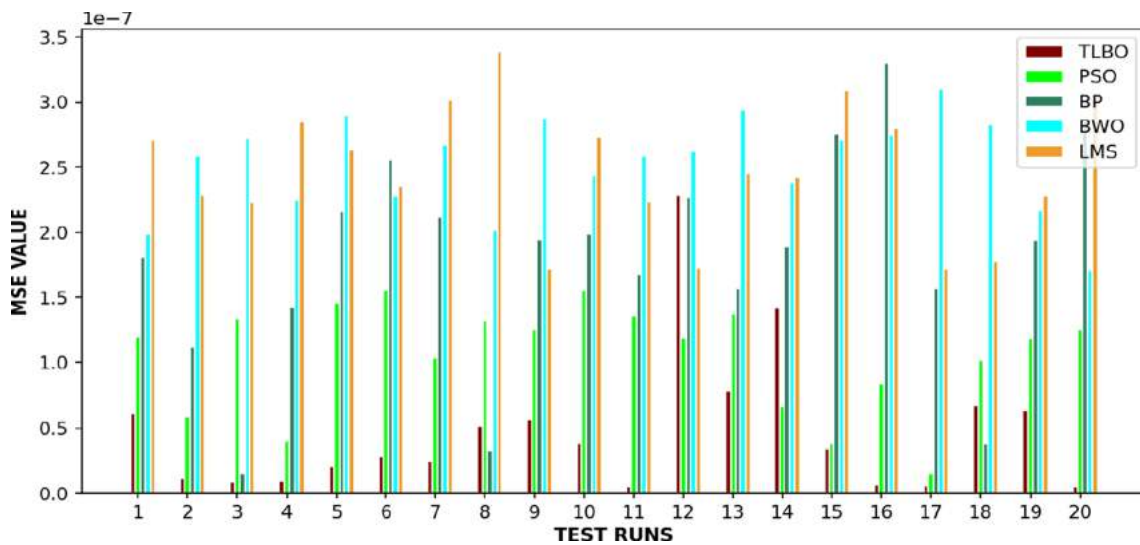


Fig. 20 Comparative plot of MSE in various test runs

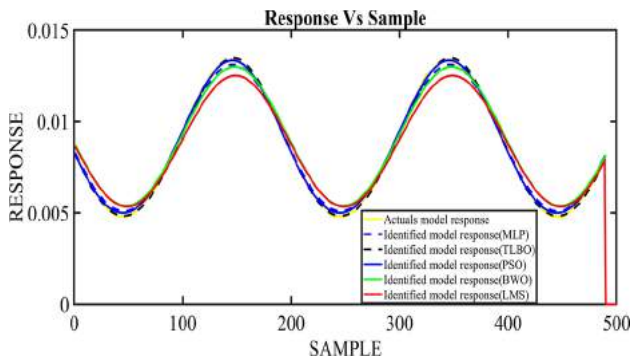


Fig. 21 Comparative identified model response

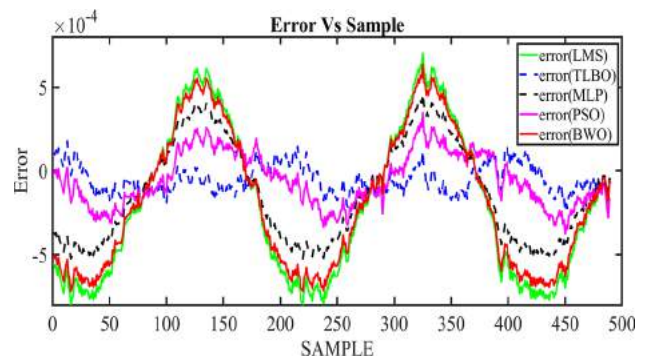


Fig. 22 Comparative error plot

and 19. The comparative results of MSE and the MSE plot are provided for various test runs in Table 5 and Fig. 20.

Here, N_i is the number of iteration, P_s is the number of population, F_e is the number of feature and N_t is the number of input for training. From the Big O Notation, it shows that the FLANN-LMS and MLP-BP have less time complexity than the other three algorithms as shown in Table 4. For investigating the performance objectively, the Mean Squares Error (MSE) is considered as the performance metric. The average values of MSE for all the five models by running them for 20 independent test runs are shown in Table 4. The MSE values for all the models, for each test run in histogram, are shown in Fig. 20. It is clear from Table 4 that the MSE values of the FLANN-TLBO algorithm are lowest as compared to others, which signify the superior performance over the other four competitive networks.

It is depicted from Fig. 21 that the predicted value does not match with the actual output and a very large gap exists.

Hence, the performance is highly unsatisfactory for the FLANN-LMS network. There exists high nonlinearity in the data of the Maglev system, and hence, the result is highly discouraging. The results of the FLANN-TLBO are found to be the most matched one as compared to the other four networks.

From Figs. 21 and 22, it is demonstrated that the response of the FLANN-TLBO model replicates the response of the real-time Maglev system and hence it is the best among all other competitive models. The performance of the algorithm also depends on the number controlling parameters and number of steps associated with weight updation. It is because these two parameters increase the computation time and the computational complexity. From Table 4, it is observed that the FLANN-LMS and FLANN-PSO have taken 4.15 s and 782.43 s CPU time, respectively, which are the lowest and the highest values. The recently developed BWO algorithm-based FLANN network required 382.422 s. The LMS algorithm have one step weight updation with one

Fig. 23 Control of actual Maglev system and identified model using Fuzzy PID controller

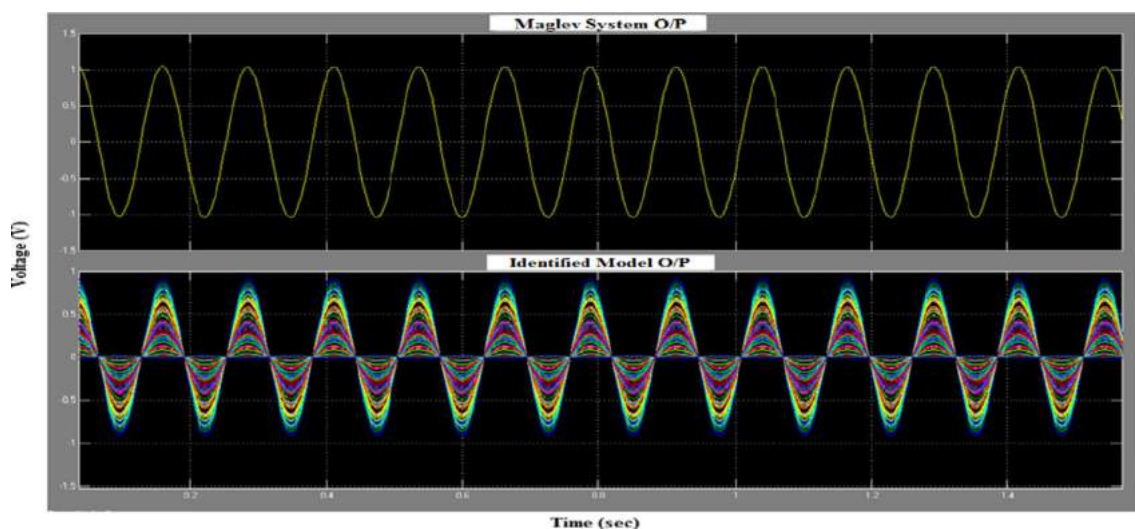
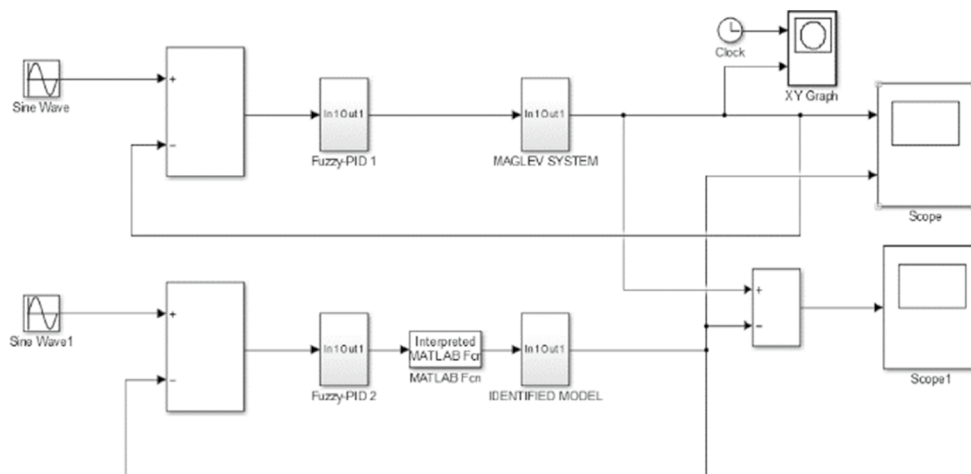


Fig. 24 Comparative results of Maglev system and identified model with a Fuzzy PID controller

Table 7 Minimum wins needed for the two-tailed sign test at $\alpha = 0.05$ and $\alpha = 0.01$

No. of cases	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
$\alpha = 0.05$	5	6	7	7	8	9	9	10	10	11	12	12	13	13	14	15	15	16	17	18	18
$\alpha = 0.01$	5	6	6	7	7	8	9	9	10	10	11	12	12	13	13	14	14	15	16	16	17

Table 8 Critical values obtained for the two-tailed sign tests at $\alpha = 0.05$ and $\alpha = 0.01$ using MSE metric as a triumphant parameter

TLBO	BWO	PSO	LMS	MLP-BP
Wins (+)	20	18	19	17
Loss (-)	0	2	1	3
Detected difference	$\alpha = 0.05$	$\alpha = 0.05$	$\alpha = 0.05$	$\alpha = 0.05$

Table 9 Sign test using MSE metric as a triumphant parameter

Comparison	<i>p</i> value	<i>h</i> value
TLBO with BWO	0.0008	1
TLBO with PSO	0.0004	1
TLBO with LMS	0.0009	1
TLBO with BP	0.0026	1

Table 10 Wilcoxon signed test using MSE metric as a triumphant parameter

Comparison	<i>P</i> value	<i>h</i> value
TLBO with BWO	0.0001	1
TLBO with PSO	0.0025	1
TLBO with LMS	0.0001	1
TLBO with BP	0.0003	1

Table 11 Friedman test rank table

Methods	BWO	TLBO	PSO	LMS	BP
Mean ranks	17	5.2	8.2	17.2	12.4

controlling parameter, the PSO have one step weight updation with three controlling parameters and the MLP have one step weight updation with three controlling parameters. The recently developed BWO algorithm have two steps weight updation with three controlling parameters for which it takes higher time and computational complexity. The FLANN-TLBO model involves two-updation process during the teaching and the learning phases, and hence, it takes more time of 462.02 s.

7.2 Validation of Identified Model

The best identified FLANN-TLBO model is chosen for the validation purpose. The identified Maglev system obtained from the optimal 90 weights is shown in Table 6. This model is controlled and validated by using Fuzzy PID controller with proper choice of the controller parameters. The FLANN-TLBO model having the optimal set of weights is shown in Table 6. The Fuzzy PID controller is used to control both the actual Maglev and the identified model to investigate the response as shown in Fig. 23.

The range of membership functions of the Fuzzy PID controller is defined from -1 to 1 , as shown in Figs. 8 and 9. The best responses are obtained after proper tuning of the Fuzzy PID controller with the values of k_p, k_i and k_d are $-4, -2$ and -0.2 . From Fig. 24, it has been observed that the Fuzzy PID-controlled identified model and the actual Maglev system exhibit the same response.

Table 12 Friedman test parameter

Source	Sum of square (SS)	Degree of freedom (DOF)	Mean square (MS)	Chi-square	Critical value (<i>p</i>)
Columns	141.1	4	35.275	56.44	1.6214E-11
Error	58.9	76	0.775		
Total	200	99			

7.3 Nonparametric Statistical Tests

To validate the dominance of the FLANN-TLBO network, the pairwise sign test and Wilcoxon signed-rank test are carried out. In fact, the sign test and the Wilcoxon signed-rank test are two well-known nonparametric statistical tests proposed for pairwise comparison of the two heuristics approaches. Here, we have carried out the test for 20 runs of each algorithm to justify a fair comparison. The results are listed in Table 8 by considering the average value of MSE as the winning parameter. The minimum number of wins required to obtain $\alpha = 0.05$ and $\alpha = 0.01$ levels of significance for one algorithm over another is shown in Table 7. It is observed in Table 5 that the FLANN-TLBO model shows dominance over all the three other models with a significance of $\alpha = 0.05$.

It is observed from the performance measures of Table 8 that the TLBO shows a significant improvement over the BWO, PSO, LMS and MLP-BP algorithms with a level of significance $\alpha = 0.05$ by taking the detection rate as the winning parameter, and the *p* value and *h* value for the sign test using the MSE metric as a triumphant parameter listed in Table 9. The *p* value and *h* value represent the superiority of the algorithm over the other competitive algorithms. If the *p* value is less than the level of significance $\alpha = 0.05$ and *h* value is 1, then the proposed algorithm is superior over the other and the null hypothesis can be rejected. If the *p* value is greater than the level of significance $\alpha = 0.05$ and *h* value is 0, then the proposed algorithm is considered to be inefficient. Similarly, the Wilcoxon signed-rank test, which is similar to the paired *t* test in statistical procedure, and normally applied to detect the dominance behaviors between the two algorithms, is also performed. The performance comparison of all the algorithms is listed in Table 10. The results presented in Tables 9 and 10 reveal the superiority of the TLBO over other competitive algorithms.

To study the supremacy and repeatability of the obtained response of the network a nonparametric Friedman test is also performed by using the MATLAB. Table 11 shows the average rank of the different networks used for identification, which signifies that the lower rank networks have higher accuracy and performance. The Friedman test parameters are given in Table 12, and the critical value is obtained as $1.6214E-11$ from the Friedman test. A null hypothesis concept emerges, if the critical value is less than the significance

level, i.e., $\alpha = 0.05$ and it can be rejected. Hence, the dominance of the proposed algorithm over other competitive algorithms has been confirmed by performing the sign test, Wilcoxon signed-rank test and Friedman test. A null hypothesis concept comes, if the critical value (p value) is less than the significance level, i.e., $\alpha = 0.05$ and it can be rejected. The result obtained reveals the supremacy of the TLBO over the others.

8 Conclusion

This article proposed a FLANN-TLBO model that yields improved identification and implementation of the Maglev plant. The performance of the TLBO-based FLANN model is compared with that of the other ANN-based models, i.e., MLP-BP, FLANN-LMS, FLANN-PSO and FLANN-BWO. The estimated models have been compared in terms of MSE, CPU time, and the response matching capability of the Maglev system. From the simulations, it is perceived that the proposed FLANN-TLBO model provides superior identification model of the actual Maglev system. The validation of the proposed FLANN-TLBO model is carried out by comparing its performance with the actual Maglev system under identical conditions. The results demonstrate improved response matching of the identified model and the actual system. Moreover, the statistical tests validate the dominance of the FLANN-TLBO network over others. The outcomes of statistical testing reveal the supremacy of the TLBO algorithm in comparison with other competitive algorithms with a significance level of $\alpha = 0.05$. Further, other variants of the neural network and nature-inspired algorithm can be applied for achieving better models of complex systems.


References

- Padoan, A.; Astolfi, A.: Nonlinear system identification for autonomous systems via functional equations methods. In: American Control Conference, pp. 1814–1819 (2016)
- Subudhi, B.; Ieee, S.M.; Jena, D.: Nonlinear system identification of a twin rotor MIMO system. In: IEEE TENCON, pp. 1–6 (2009)
- Weng, B.; Barner, K.E.: Nonlinear system identification in impulsive environments. *IEEE Trans. Signal Process.* **53**, 2588–2594 (2005)
- Forrai, A.: System identification and fault diagnosis of an electromagnetic actuator. *IEEE Trans. Control Syst. Technol.* **25**, 1028–1035 (2017). <https://doi.org/10.1109/TCST.2016.2582147>
- Mondal, A.; Sarkar, P.: A unified approach for identification and control of electro-magnetic levitation system in delta domain. In: International Conference on Control, Instrumentation, Energy and Communication, pp. 314–318 (2016)
- Srivatava, S.; Gupta, M.: A novel technique for identification and control of a non linear system. In: International Conference on Computational Intelligence and Networks, pp. s172–176 (2016). <https://doi.org/10.1109/CINE.2016.37>
- Wen, S.; Wang, Y.; Tang, Y.; Xu, Y.; Li, P.; Zhao, T.: Real-time identification of power fluctuations based on LSTM recurrent neural network: a case study on Singapore Power System. *IEEE Trans. Ind. Inf.* **15**, 5266–5275 (2019). <https://doi.org/10.1109/tii.2019.2910416>
- Alqahtani, A.; Marafi, S.; Musallam, B.; El, N.; Abd, D.; Khalek, E.: Photovoltaic Power Forecasting Model Based on Nonlinear System Identification Modèle de prévision de puissance photovoltaïquebasésur l'. *identification de système non-linéaire* **39**, 243–250 (2016)
- Nanda, S.J.; Panda, G.; Majhi, B.: Improved identification of nonlinear dynamic systems using artificial immune system. In: IEEE Conference and Exhibition on Control, Communications and Automation, pp. 268–273 (2008). <https://doi.org/10.1109/INDCON.2008.4768838>
- Patra, J.; Pal, R.; Chatterji, B.N.; Panda, G.: Identification of nonlinear dynamic systems using functional link artificial neural networks. *IEEE Trans. Syst.* **29**, 254–262 (1999)
- Majhi, B.; Panda, G.: Robust identification of nonlinear complex systems using low complexity ANN and particle swarm optimization technique. *Expert Syst. Appl.* **38**, 321–333 (2011). <https://doi.org/10.1016/j.eswa.2010.06.070>
- Han, M.: Robust Structure Selection of Radial Basis Function Networks for Nonlinear System Identification (2014)
- Chen, W.: Nonlinear System Identification Based on Radial Basis Function Neural Network Using Improved Particle Swarm Optimization, pp. 409–413 (2009). <https://doi.org/10.1109/ICNC.2009.233>
- Kumpati, S.N.; Kannan, P.: Identification and control of dynamical systems using neural networks. *IEEE Trans. Neural Netw.* **1**, 4–26 (1990)
- Pao, Y.H.: *Adaptive Pattern Recognition and Neural Networks*. Addison-Wesley, Reading (1989)
- Mallikarjuna, B.; Viswanathan, R.; Naib, B.B.: Feedback-based gait identification using deep neural network classification. *J. Crit. Rev.* **7**, 661–667 (2020). <https://doi.org/10.31838/jcr.07.04.125>
- Vora, D.R.; Rajamani, K.: A hybrid classification model for prediction of academic performance of students: a big data application. *Evol. Intell.* (2019). <https://doi.org/10.1007/s12065-019-00303-9>
- Guo, Y.; Wang, F.; Lo, J.T.H.: Nonlinear system identification based on recurrent neural networks with shared and specialized memories. In: Asian Control Conference 2018-January, pp. 2054–2059 (2018). <https://doi.org/10.1109/ASCC.2017.8287491>
- Wang, Z.; Gu, H.: Nonlinear system identification based on genetic algorithm and grey function. In: IEEE International Conference on Automation and Logistics, pp. 1741–1744 (2007)
- Guoqiang, Y.; Weiguang, L.; Hao, W.: Study of RBF neural network based on PSO algorithm in nonlinear system (2015). <https://doi.org/10.1109/ICICTA.2015.217>
- Kang, D.; Lee, B.; Won, S.: Nonlinear system identification using ARX and SVM with advanced PSO. In: IEEE Industrial Electronics Society, pp. 598–603 (2007)
- Panda, G.; Mohanty, D.; Majhi, B.; Sahoo, G.: Identification of nonlinear systems using particle swarm optimization technique. In: IEEE Congress on Evolutionary Computation, pp. 3253–3257 (2007). <https://doi.org/10.1109/CEC.2007.4424889>
- Hayyolalam, V.; PourhajiKazem, A.A.: Black widow optimization algorithm: a novel meta-heuristic approach for solving engineering optimization problems. *Eng. Appl. Artif. Intell.* **87**, 103249 (2020). <https://doi.org/10.1016/j.engappai.2019.103249>
- Kumar, M.; Kulkarni, A.J.; Satapathy, S.C.: Socio evolution & learning optimization algorithm: a socio-inspired optimization



- methodology. *Future Gener. Comput. Syst.* **81**, 252–272 (2018). <https://doi.org/10.1016/j.future.2017.10.052>
25. Gholizadeh, S.; Milany, A.: An improved fireworks algorithm for discrete sizing optimization of steel skeletal structures. *Eng. Optim.* **50**, 1829–1849 (2018). <https://doi.org/10.1080/0305215X.2017.1417402>
 26. Gholizadeh, S.; Ebadijalal, M.: Performance based discrete topology optimization of steel braced frames by a new metaheuristic. *Adv. Eng. Softw.* **123**, 77–92 (2018). <https://doi.org/10.1016/j.advengsoft.2018.06.002>
 27. Gholizadeh, S.; Danesh, M.; Gheytratmand, C.: A new Newton metaheuristic algorithm for discrete performance-based design optimization of steel moment frames. *Comput. Struct.* **234**, 106250 (2020). <https://doi.org/10.1016/j.compstruc.2020.106250>
 28. Rao, R.V.; Savsani, V.J.; Vakharia, D.P.: Teaching-learning-based optimization: a novel method for constrained mechanical design optimization problems. *Comput. Aided Des.* **43**, 303–315 (2011). <https://doi.org/10.1016/j.cad.2010.12.015>
 29. Naik, B.; Nayak, J.; Behera, H.S.: A TLBO based gradient descent learning-functional link higher order ANN: An efficient model for learning from non-linear data. *J. King Saud Univ. Comput. Inf. Sci.* **30**, 120–139 (2018). <https://doi.org/10.1016/j.jksuci.2016.01.001>
 30. Naumovi, M.B.; Veseli, B.R.: Magnetic levitation system in control engineering education. *Autom. Control Robot.* **7**, 151–160 (2008)
 31. Morales, R.; Feliu, V.; Member, S.; Sira-ramírez, H.; Member, S.: Nonlinear control for magnetic levitation systems based on fast online algebraic identification of the input gain. *IEEE Trans. Control Syst. Technol.* **19**, 757–771 (2011)
 32. Balko, P.; Rosinova, D.: Modeling of magnetic levitation system. In: *International Conference on Process Control*. pp. 252–257 (2017)
 33. Liceaga-castro, J.; Hernandez-alcantara, D.; Amezcua-brooks, L.: Nonlinear control of a magnetic levitation system. In: *Electronics, Robotics and Automotive Mechanics Conference*, pp. 391–396 (2009). <https://doi.org/10.1109/CERMA.2009.10>
 34. Magnetic Levitation: Control Experiments Feedback Instruments Limited (2011)
 35. Ghosh, A.; Krishnan, T.R.; Tejaswy, P.; Mandal, A.; Pradhan, J.K.; Ranasingh, S.: Design and implementation of a 2-DOF PID compensation for magnetic levitation systems. *ISA Trans.* **53**, 1216–1222 (2014)
 36. Swain, S.K.; Sain, D.; Kumar, S.; Ghosh, S.: Real time implementation of fractional order PID controllers for a magnetic levitation plant. *Int. J. Electron. Commun.* **78**, 141–156 (2017)
 37. Yaghini, M.; Khoshraftar, M.M.; Fallahi, M.: A hybrid algorithm for artificial neural network training. *Eng. Appl. Artif. Intell.* **26**(1), 293–301 (2013)
 38. Patra, J.C.; Kot, A.C.: Nonlinear dynamic system identification using Chebyshev functional link artificial neural networks. *IEEE Trans. Syst. Man. Cybern.* **32**, 505–511 (2002). <https://doi.org/10.1109/TSMCB.2002.1018769>
 39. Subudhi, B.; Jena, D.: Nonlinear system identification using memetic differential evolution trained neural networks. *Neurocomputing.* **74**, 1696–1709 (2011). <https://doi.org/10.1016/j.neucom.2011.02.006>
 40. Katari, V.; Malireddi, S.; Satya, S.K.; Panda, G.: Adaptive nonlinear system identification using comprehensive learning PSO. In: *International Symposium on Communications, Control and Signal Processing*, pp. 434–439 (2008). <https://doi.org/10.1109/ISCCSP.2008.4537265>
 41. Juang, J.-G.; Lin, B.-S.: Nonlinear system identification by evolutionary computation and recursive estimation method. In: *American Control Conference*, pp. 5073–5078 (2005). <https://doi.org/10.1109/CINE.2015.22>
 42. Puchta, E.D.P.; Siqueira, H.V.; Kaster, M.D.S.: Optimization tools based on metaheuristics for performance enhancement in a gaussian adaptive PID controller. *IEEE Trans. Cybern.* **50**, 1185–1194 (2020). <https://doi.org/10.1109/TCYB.2019.2895319>
 43. Rao, R.V.; Savsani, V.J.; Balic, J.: Teaching-learning-based optimization algorithm for unconstrained and constrained real-parameter optimization problems. *Eng. Optim.* **44**, 1447–1462 (2012). <https://doi.org/10.1080/0305215X.2011.652103>
 44. Kumar, M.; Mishra, S.K.: Teaching learning based optimization-functional link artificial neural network filter for mixed noise reduction from magnetic resonance image. *Bio Med. Mater. Eng.* **28**, 643–654 (2017)
 45. Singh, S.; Ashok, A.; Kumar, M.; Rawat, T.K.: Adaptive infinite impulse response system identification using teacher learner based optimization algorithm. *Appl. Intell.* **49**, 1785–1802 (2018). <https://doi.org/10.1007/s10489-018-1354-4>
 46. Patra, J.C.; Kot, A.C.: Nonlinear dynamic system identification using chebyshev functional link artificial neural network. In: *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 32, pp. 505–511 (2002)
 47. Li, M., He, Y.: Nonlinear system identification using adaptive Chebyshev neural networks. *IEEE International Conference on Intelligent Computing and Intelligent Systems*, pp. 243–247 (2010)
 48. Nanda, S.J., Panda, G., Majhi, B., Tah, P.: Improved Identification of Nonlinear MIMO Plants using New Hybrid FLANN-AIS Model. In: *International Advanced Computing Conference*, pp. 141–146 (2009). <https://doi.org/10.1109/IADCC.2009.4808996>
 49. Kumar, M.; Mishra, S.K.: Particle swarm optimization-based functional link artificial neural network for medical image denoising. In: *Computational Vision and Robotics*, vol. 105–111 (2015)
 50. Arora, A.; Hote, Y.V.; Rastogi, M.: Design of PID controller for unstable system. *Commun. Comput. Inf. Sci.* **140**, 19–26 (2011). https://doi.org/10.1007/978-3-642-19263-0_3
 51. Rastogi, M.A.; Arora, Y.V.H.: Design of Fuzzy Logic Based PID Controller for an Unstable System, Vol. 157, p. 66–571. Springer, Berlin (2011)



ITC 3/50 Information Technology and Control Vol. 50 / No. 3 / 2021 pp. 411-423 DOI 10.5755/j01.itc.50.3.27349	Development of Proposed Ensemble Model for Spam e-mail Classification	
	Received 2020/07/18	Accepted after revision 2021/06/08
	 http://dx.doi.org/10.5755/j01.itc.50.3.27349	

HOW TO CITE: Shrivas, A. K., Dewangan, A. K., Ghosh, S. M., Singh, D. (2021). Development of Proposed Ensemble Model for Spam e-mail Classification. *Information Technology and Control*, 50(3), 411-423. <https://doi.org/10.5755/j01.itc.50.3.27349>

Development of Proposed Ensemble Model for Spam e-mail Classification

Akhilesh Kumar Shrivas

Guru Ghasidas Vishwavidyalaya, Bilaspur (C.G.), India

Amit Kumar Dewangan

Dr. C. V. Raman University, Bilspur (C.G.), India

S. M. Ghosh

Dr. C. V. Raman University, Bilspur (C.G.), India

Devendra Singh

Guru Ghasidas Vishwavidyalaya, Bilaspur (C.G.), India


Corresponding author: akhilesh.mca29@gmail.com

Spam e-mail documents classification is a very challenging task for e-mail users, especially non IT users. Billions of people using the internet and face the problem of spam e-mails. The automatic identification and classification of spam e-mails help to reduce the problem of e-mail users in managing a large amount of e-mails. This work aims to do a significant contribution by building a robust model for classification of spam e-mail documents using data mining techniques. In this paper, we use Enorn1 data set which consists of spam and ham documents collected from Kaggle repository. We propose an Ensemble Model-1 that is an ensemble of Multilayer Perceptron (MLP), Naïve Bayes and Random Forest (RF) to obtain better accuracy for the classification of spam and hame-mail documents. Experimental results reveal that the proposed Ensemble Model-1 outperforms other existing classifiers as well as other proposed ensemble models in terms of classification accuracy. The suggested and proposed Ensemble Model-1 produces a high accuracy of 97.25% for classification of spam e-mail documents.

KEYWORDS: Ensemble Model, Classification, Data Mining, Spam e-mail, Machine Learning.

Single-source precursors for main group metal sulfides and solar cell applications

Running head: Single-source precursors

 The corrections made in this section will be reviewed and approved by master copier.

Suryabhan Singh^a, Ashish Kumar Singh^a, Abhinav Kumar^b

Q8 ^aDepartment of Chemistry, Guru Ghasidas Vishwavidyalaya, Bilaspur, India

^bDepartment of Chemistry, University of Lucknow, Lucknow, India

Keywords:

No keywords are available

11.1 Introduction

Metal sulfides are a class of substances known for their unique chemical and physical properties [1–3]. Their semiconducting nature means that they have potential in various applications. Although a variety of methods are known for the syntheses of metal sulfides, traditional methods such as solid-state reaction, chemical vapor deposition (CVD), and homogeneous precipitation methods suffer from difficulties such as temperature requirements, long reaction times, formation of impure products, mixed phases, etc. [1–6]. Compared to bulk materials, thin films of main group metal sulfides play a critical role in electronic applications such as solar cells, optical fiber communications, and full-color displays. The purity of precursors is a crucial requirement in order to limit contamination in deposited thin films [6–9]. An alternative and energy-efficient method for the synthesis of thin films is the single molecular source precursor or single-source precursor (SSP) approach where the organic/metal fragments present in the metal complexes of sulfur-containing ligands are removed and pure phase metal sulfides are reassembled under mild conditions [Instruction: reference span should be [1-10] instead of [1-9].] [1–9].

Dual/multiple-source precursor approaches are traditionally involved in the synthesis of bulk/thin films of metal sulfides, however, single-source precursors have been suggested as a method for lowering the growth temperature. The desired synthetic reaction pathway involves adsorption of the precursor without breaking the core M-S bond but with loss of the ancillary ligands. Single-source precursors potentially display a range of advantages over conventional CVD precursors, such as limited pre-reaction owing to only one precursor, reduced toxicity, possible lower temperature growth as well as possible stability in air and moisture. However, commercialization of single-source precursors is still a challenge due to their disadvantages such as difficulties in control of stoichiometry of the final product, growth of ternary and quaternary materials, low volatility, and prevention of epitaxial growth of polynuclear decomposition fragments due to low surface mobility. Herein, we present the various methods for the synthesis of bulk materials and thin films of main group metal sulfides from single-source precursor and their use for application in solar cells.

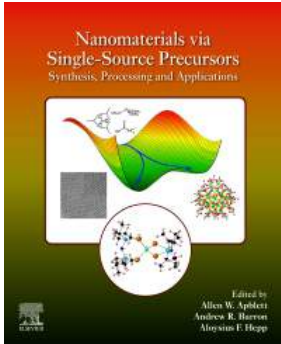
11.2 Synthetic methods

The essential requirements for a suitable precursor are synthetic ease, ability to be synthesized in good yield and purity, high volatility, stability under ambient conditions, and decomposition at elevated temperatures. Purity requires



Home (<https://www.elsevier.com>) > Physical Sciences and Engineering (<https://www.elsevier.com/physical-sciences-and-engineering>)
 > Materials Science (<https://www.elsevier.com/physical-sciences-and-engineering/materials-science>)
 > Books (<https://www.elsevier.com/physical-sciences-and-engineering/materials-science/books>) > Nanomaterials via Single-Source Precursors
 Back to search results (<https://www.elsevier.com/search-results?labels=books&query=single%20source%20precursor&page=1>)

Happy Holidays!: 25% off print and eBooks. No promo code needed. Terms & conditions (<https://www.elsevier.com/books-and-journals/special-offers>)



Nanomaterials via Single-Source Precursors Synthesis, Processing and Applications

1st Edition - January 17, 2022

Editors: Allen Apblett, Andrew Barron, Aloysius Hepp

Paperback ISBN: 9780128203408

Purchase options

Select country/region India

Print - Paperback **25% off**

~~\$275.00~~ **\$206.25**

Available for Pre-Order [i](#)

Pre-Order

Sales tax will be calculated at check-out

Institutional Subscription

[Request a Sales Quote](#)

Tax Exempt Orders [Support Center \(https://service.elsevier.com/app/answers/detail/a_id/7122/supporthub/ecommerce/p/12751/\)](https://service.elsevier.com/app/answers/detail/a_id/7122/supporthub/ecommerce/p/12751/)

Returns & Refunds [\(https://service.elsevier.com/app/answers/list/c/10539/supporthub/ecommerce/\)](https://service.elsevier.com/app/answers/list/c/10539/supporthub/ecommerce/)



Free Global Shipping

No minimum order

50% off Book Bundles

Immediately download your ebook while waiting for print delivery

No promo code needed

[More Details \(https://www.elsevier.com/books-and-journals/special-offers\)](https://www.elsevier.com/books-and-journals/special-offers)

Description

Nanomaterials via Single-Source Precursors: Synthesis, Processing and Applications presents recent results and overviews of synthesis, processing, characterization and applications of advanced materials for energy, electronics, biomedicine, sensors and aerospace. A variety of processing methods (vapor, liquid and solid-state) are covered, along with materials, including metals, oxides, semiconductor, sulfides, selenides, nitrides, and carbon-based materials. Production of quantum dots, nanoparticles, thin films and composites are described by a collection of international experts. Given the ability to customize the phase morphology, and properties of target materials, this “rational approach” to synthesis and processing is a disruptive technology for electronic, energy, structural and biomedical (nano)materials and devices. The use of single-source chemical precursors for materials processing technology allows for intimate elemental mixing and hence



production of complex materials at temperatures well below traditional physical methods and those involving direct combination of elements. The use of lower temperatures enables thin-film deposition on lightweight polymer substrates and reduces damage to complex devices structures such as used in power, electronics and sensors.

Key Features 

Readership 

Table of Contents 

Part 1: Synthesis and characterization of single source precursors

1. Precursor design and impact of structure on the fabrication of materials
2. Structural studies of main group organometallic single source precursors for MOCVD
3. Dithiocarbamate complexes containing the pyrrole moiety for synthesis of sulfides
4. Theoretical studies of gas-phase decomposition of single source precursors

Part 2: Processing of single-source precursors into materials

5. Semiconductor clusters and their use as single source precursors to nanomaterials
6. Chalcogenoethers as convenient synthons for low-temperature solutionphase synthesis of metal chalcogenide nanocrystals
7. Synthesis of lanthanide chalcogenide nanoparticles
8. Organometallic single-source precursors to zinc oxide-based nanomaterials
9. Nickel chalcogenide thin films and nanoparticles from molecular single-source precursors

Part 3: Single-source precursor-derived materials for energy conversion and catalysis

10. Group 15/16 Single-Source Precursors for energy materials
11. Single-source precursors for main group metal sulfides and solar cell applications
12. Fabrication and catalytic applications of first row-transition metal and mixed-metal chalcogenides synthesized from single-source precursors
13. Single-source heterometallic precursors to MOCVD Pd Cu alloy films for energy and catalysis applications

Part 4: Materials from single-source precursors: Applications and practical considerations

15. Zinc acetate amine complexes: Single-source precursors to zinc oxide films and nanoparticles; the influence of amines on photocatalysis
16. Coinage metal chalcogenides via single-source precursors
17. Commercialization of single-source precursors: Applications, intellectual property, and technology transfer

Product details 

About the Editors 

Solutions

Scopus (/solutions/scopus)

ScienceDirect (/solutions/sciencedirect)

Mendeley (/solutions/mendeley)

Evolve (/solutions/evolve)

Knovel (/solutions/knovel-engineering-information)

Reaxys (/solutions/reaxys)

ClinicalKey (/solutions/clinicalkey)

Search by keyword, title, subject area

Researchers

Submit your paper (/authors/journal-authors/submit-your-paper)



[Visit Author Hub \(/authors\)](#)

[Visit Editor Hub \(/editors\)](#)

[Visit Librarian Hub \(/librarians\)](#)

[Visit Reviewer Hub \(/reviewers\)](#)

Subjects

[Health \(/health\)](#)

[Life Sciences \(/life-sciences\)](#)

[Physical sciences & engineering \(/physical-sciences-and-engineering\)](#)

[Social sciences & humanities \(/social-sciences-and-humanities\)](#)

About Elsevier

[About \(/about\)](#)

[Careers \(/about/careers\)](#)

[Newsroom \(/about/newsroom\)](#)

[Events \(/events\)](#)


[Publisher relations \(/publisher-relations\)](#)

[Advertising, reprints and supplements \(/advertising-reprints-supplements\)](#)

How can we help?

[Support and Contact \(https://www.elsevier.com/support\)](https://www.elsevier.com/support)

Select location/language

 [Global - English\(/location-selector\)](#)



<https://www.elsevier.com>

ELSEVIER

Copyright © 2022 Elsevier, except certain content provided by third parties

Cookies are used by this site. [Cookie Settings](#)

[Terms and Conditions \(/legal/elsevier-website-terms-and-conditions\)](#)
[Privacy Policy \(/legal/privacy-policy\)](#)
[Cookie Notice \(/legal/cookie-notice\)](#)
[Sitemap \(/sitemap\)](#)



<https://www.elsevier.com>

 **RELX™** (<https://www.relx.com/>)

ELSEVIER

 **RELX™** (<https://www.relx.com/>)

f	in	tw	yt
(ht	(ht	(ht	(ht
tp	tp	tp	tp
s:/	s:/	s:/	s:/
/w	/w	/t	/w
w	w	wi	w
w.	w.l	tte	w.
fa	in	r.c	yo
ce	ke	o	ut
bo	di	m	ub
ok	n.	/El	e.c
/c	co	se	o
o	m	vi	m
m	/c	er	/c/
/El	o	C	els
se	m	on	ev
vi	pa	ne	ier
er	ny	ct)	/)
C	/el		
on	se		
ne	vi		
ct)	er)		