


RESEARCH

Open Access



Elitist-opposition-based artificial electric field algorithm for higher-order neural network optimization and financial time series forecasting

Sarat Chandra Nayak^{1*} , Satchidananda Dehuri² and Sung-Bae Cho¹

*Correspondence:
saratnayak234@gmail.com;
saratnayak234@yonsei.ac.kr

¹ Soft Computing Laboratory,
Department of Computer
Science, Yonsei University, 50
Yonsei-ro, Sudaemoon-gu,
Seoul 03722, South Korea

² Department of Computer
Science, Fakir Mohan University,
Balasore, Odisha 756019, India

Abstract

This study attempts to accelerate the learning ability of an artificial electric field algorithm (AEFA) by attributing it with two mechanisms: elitism and opposition-based learning. Elitism advances the convergence of the AEFA towards global optima by retaining the fine-tuned solutions obtained thus far, and opposition-based learning helps enhance its exploration ability. The new version of the AEFA, called elitist opposition leaning-based AEFA (EOAEFA), retains the properties of the basic AEFA while taking advantage of both elitism and opposition-based learning. Hence, the improved version attempts to reach optimum solutions by enabling the diversification of solutions with guaranteed convergence. Higher-order neural networks (HONNs) have single-layer adjustable parameters, fast learning, a robust fault tolerance, and good approximation ability compared with multilayer neural networks. They consider a higher order of input signals, increased the dimensionality of inputs through functional expansion and could thus discriminate between them. However, determining the number of expansion units in HONNs along with their associated parameters (i.e., weight and threshold) is a bottleneck in the design of such networks. Here, we used EOAEFA to design two HONNs, namely, a pi-sigma neural network and a functional link artificial neural network, called EOAEFA-PSNN and EOAEFA-FLN, respectively, in a fully automated manner. The proposed models were evaluated on financial time-series datasets, focusing on predicting four closing prices, four exchange rates, and three energy prices. Experiments, comparative studies, and statistical tests were conducted to establish the efficacy of the proposed approach.

Keywords: AEFA, Elitism, Opposition-based learning, Improved AEFA, HONN, PSNN, FLANN, Financial forecasting

Introduction

Financial time series (FTS) comprise stock market, commodity, energy, gold, silver, and currency exchange prices, etc., whose behaviors are uncertain and sensitive to various international aspects and socio-economic factors (Hsu et al. 2016). The inherent dynamism, nonlinearities, and non-stationarity of such FTS data make

the prediction process challenging (Fama 1970; Kara et al. 2011). As a complex and dynamic system, FTS forecasting requires an accurate prediction model and has been an attractive research area in the domains of data mining and financial engineering. Statistical methods were early approaches of FTS forecasting (Zhang 2003; Adhikari and Agrawal 2014; Box et al. 2015). However, their approximations are deprived and mostly fail to model the underlying dynamism and replicate the ever-changing patterns of FTS (Akbulgic et al. 2014; Nayak and Misra 2019a; Das et al. 2022; Nayak 2022).

Recently, nonlinear approximation systems such as artificial neural networks (ANNs) and deep-learning-based forecasting have been applied in FTS forecasting which are data-driven in nature and have established their predictability by exploring huge amounts of available historical financial data (Nayak 2022; Akman et al. 2020). Typically, ANNs comprise a multi-layered architecture, providing a “black box” visualization. Multiple layers lead to multiple weight and threshold updates, an elongated convergence rate, and being trapped in local optima owing to gradient descent-based learning, thereby attracting experts to frame simple and flat neural models. Higher-order neural networks (HONNs) are a class of feed-forward ANN capable of providing nonlinear decision margins and achieving an improved classification accuracy compared with linear neurons (Guler and Sahin 1994). The introduction of higher-order terms differentiates them from normal ANNs. Unlike summing elements in conventional ANNs, HONNs use both summing elements and the products of weighted inputs, which are called higher-order terms. Because of their single layer and smaller number of adjustable model parameters, they have simple and flat architectures. They can capture nonlinearities coupled with complex real-world data using a simple flat architecture (Shin and Ghosh 1995; Park et al. 2000). The representational power of these higher-order terms helps enhance the information capacity of a model in solving nonlinear problems with a smaller network and faster convergence rate (Leerink et al. 1994). Altogether, HONNs exhibit fast learning, a robust approximation with fault tolerance competence, and influential input–output mapping with solitary trainable weights and thresholds (Wang et al. 2008). The functional link artificial neural network (FLANN) proposed by Pao (1992) and pi-sigma neural network (PSNN) proposed in Shin and Ghosh (1991) are popular HONNs applied in many areas of engineering optimization. Financial forecasting using HONNs has been determined to be better than that using conventional multilayer ANNs (Das et al. 2020; Ghazali 2005; Nayak et al. 2015). To alleviate the limitations of traditional ANNs, higher order product unit neural networks (HPUNNs) (Giles and Maxwell 1987) and product unit neural networks (PUNNs) (Durbin and Rumelhart 1989) have been proposed in the literature with amended nonlinear mapping abilities. However, an exponential increase in higher-order terms increases the magnitude and complexity of the networks. To better control the amplified model parameters and processing units, a PSNN with a robust classification ability was suggested by Shin and Ghosh (1991). A PSNN was proposed in Nayak et al. (2015) for successful prediction of future stock indices. An HONN with a Bayesian confidence measure was proposed by Knowles et al. for EUR/USD exchange rate forecasting (Knowles et al. 2005). A FLANN with trigonometric basis functions and recursive LMS algorithm for weight updating were developed for S&P 500 and DJIA index forecasting (Majhi et al. 2009).

The network magnitude and learning method significantly influence the performance of an ANN. The most widely used learning method in ANNs (gradient descent learning) has been victimized by sluggish convergence speeds, inaccurate learning, leaning towards local optima, and augmented computational overhead. To address these problems, a few nature-inspired learning algorithms have been developed and extensively used for training ANNs. Although such methods are used to train ANNs to solve multifaceted problems, their proficiency is mostly determined through well-modified control parameters. The selection of such learning parameters while designing a model necessitates human intervention and domain expertise, making the method difficult to use. Typically, for a particular problem, the selection of such parameters requires numerous trial-and-error steps, and an improper assortment leads the search operation towards a local optimum, thereby generating an inaccurate solution. Hence, techniques with fewer control parameters and sophisticated approximation capabilities are of interest.

To adjust ANN parameters, evolutionary algorithms such as particle swarm optimization (PSO), genetic algorithm (GA), and differential evolution (DE) have been demonstrated to be proficient (Shadbolt 2004). As no solitary apt method has been developed for resolving all classes of problems, incessant perfections are ongoing through the enhancement (Opara and Arabas 2018; Jiang et al. 2014) or hybridization (Nayak and Misra 2019b; Nayak and Ansari 2019a; Chiroma et al. 2015) of existing methods. Recently, the artificial electric field algorithm (AEFA) was projected as an optimization procedure based on the principle of electrostatic force (Yadav 2019; Yadav and Kumar 2020, 2019). The conceptualization of the AEFA is based on the theoretical perception of electric fields, charged particles, and the force of attraction/repulsion. The mathematical representation of its learning ability, acceleration update, and convergence speed has been well demonstrated in the literature by answering benchmark optimization tasks (Yadav 2019; Yadav and Kumar 2020, 2019).

Metaheuristic algorithms are stochastic in nature and their performance and efficiency vary from one dataset to another. Several schemes have been proposed to modify existing metaheuristic algorithms to improve their accuracy. The oppositional-based learning (OBL) concept has been proposed to enhance the performance of machine learning algorithms (Tizhoosh 2005). Several attempts have been made to establish theoretical extensions, incorporating OBL with existing metaheuristics to improve their performance as well as real-world engineering applications. A systematic survey was conducted in Mahdavi et al. (2018), summarizing the use and growth of OBL.

In this study, we designed an improved optimization technique by incorporating the concepts of elitism and OBL into the basic AEFA, called elitist-opposition-based AEFA (EOAEFA). Elitism advances the convergence of the AEFA towards global optima by retaining the finest solutions obtained thus far, and OBL helps enhance the diversification ability of the AEFA. The EOAEFA was used to explore the potential weights and thresholds of two competitive HONNs, FLANN and PSNN, independently, thereby forming two hybrid models. The performances of the two hybrid models were evaluated for predicting the futures prices of 11 dynamic and chaotic FTS, including four stock closing prices, four currency exchange, and three energies prices series. For fair comparison, five other forecasts were developed in a similar manner. The comparison models include the basic AEFA-based FLANN and PSNN (i.e., AEFA-FLN and AEFA-PSNN),

the genetic algorithm-based FLANN and PSNN (i.e., GA-FLN and GA-PSNN), and a back propagation neural network (BPNN). Training/test patterns were selected using the sliding window algorithm, data normalization was performed using the sigmoid method, and all models were adaptively trained to reduce training costs. Finally, comparative studies and statistical tests were conducted to ensure the predictive ability of EOAEFA + HONN. The original contributions of this study are as follows:

- An improved learning algorithm, EOAEFA, is proposed by integrating the OBL and elitism concepts with the basic AEFA.
- The EOAEFA was used to adjust the parameters of two HONNs; thus, two computationally efficient hybrid models were created: EOAEFA-FLN and EOAEFA-PSNN.
- The hybrid models were evaluated to predict 11 real-world FTS with a systematic performance evaluation.

This paper is organized into seven sections. Sect. "[Background and related work](#)" summarizes problem-related studies. Sect. "[Proposed EOAEFA+HONN forecasting method](#)" describes the methodologies used. Sect. "[FTS data and statistical analysis](#)" briefly discusses the FTS data and their statistics. Sect. "[Results and analysis](#)" describes and analyzes the simulation results. Sect. "[Statistical testing and further analysis](#)" discusses the significance of the models. Finally, Sect. "[Conclusions and future work](#)" presents the conclusions, limitations, and possible extensions of this study.

Background and related work

This section analyzes recent related research in the area of FTS forecasting using machine learning techniques. In addition, it explores metaheuristic learning methods, their variants, and opposition-based learning applied for ANN parameter tuning. To this end, the mathematical modelling of two HONNs, PSNN and FLANN, are discussed.

The distribution of financial data is complex because of the intricacy of human behavior and varying social environments. to the detection, scope optimization, and interpretation of clusters in large-scale financial datasets are difficult tasks. Li et al. proposed a computationally efficient integrated approach that detects a reasonable number of clusters and evaluates their quality (Li et al. 2021). Kou et al. designed a bankruptcy prediction model using payment-network-based variables and transactional data, incorporating optimal feature subset selection and importance evaluations for small- and medium-sized enterprises (Kou et al. 2021a). This approach achieved a satisfactory classification accuracy with a reduced feature subset. Researchers have evaluated five financial technology-based investments in European banking services (Kou et al. 2021b). This study identified payment and money transfer systems as the most important investment alternatives for advancing the financial performance of European banks, nourishing customer expectations, easing the banks' collection of receivables, and reducing operational outlays.

Although many conventional and statistical method-based predictive systems for FTS forecasting have been proposed, they mostly failed to describe nonlinearities coupled with huge volumes of financial data. They are less competent than machine learning (ML) methods, such as ANN, RNN, deep learning (DL), and HONNs. ANNs have been

well established for FTS forecasting. Different ML models such as random forest and gradient boosting approaches have been used by Yoon for real GDP growth forecasting (Yoon 2020). Methods such as multilayer perceptron (MLP) (Ecer et al. 2020), support vector machine (SVR) (Guo et al. 2018), and ANN (Nayak 2022; Akman et al. 2020; Shadbolt 2004) have proven successful in capturing nonlinearities coupled with FTS. However, multiple hidden layers in conventional ANNs lead to a longer convergence speed, become stuck in a local optimum owing to traditional back-propagation-based learning, and offer black-box visualization, thereby paving the path towards designing flat and simple ANNs.

To address the disadvantages of the backpropagation learning of HONNs, several evolutionary algorithms have been developed and extensively used for HONN training in the last two decades. To improve the performance of HONNs, chemical reaction optimization is used for training, and the resulting hybrid model is applied to stock market forecasting (Nayak et al. 2017a). The hybrid model has been demonstrated to be superior to other comparative methods. The HONN parameters tuning via chemical reaction optimization (CRO) is conducted for the effective prediction of exchange rate series (Nayak et al. 2017b). Adaptive HONNs trained using different evolutionary optimization techniques were proposed in Sahu et al. (2016); Nayak 2017; Nayak et al. 2016) to predict fluctuations in stock market data. Some hybrid models called RAFLANNs have been proposed and determined to be effective in terms of computational cost and accuracy (Subhranginee et al. 2021). A hybrid PSNN model was proposed to predict GOLD/INR and GOLD/AED prices and showed significant results (Dash et al. 2021a). Reference (Naik et al. 2021) developed a hybrid FLANN for classification whose accuracy was superior than that of other comparative models. To control the uncertainties associated with crude oil prices, Nayak et al. (2020) proposed a hybrid forecasting model by aggregating the generalization power of PSNN with the effective learning ability of FWA and determined that the FWA-PSNN model is superior to an MLP and other models (Nayak 2020). The outcome of a hybrid PSNN-SDE model was tested using various evaluation metrics, such as root mean squared error (RMSE), mean absolute percentage of error (MAPE), and Theil's U statistic (TU) and exhibited an enhanced forecasting accuracy (Rajashree et al. 2019). PSNN and FLANN have been hybridized with different training algorithms, such as GA, CRO, and COA, to expand their input space portrayal and high learning ability and establish significant developments in performance (Nayak and Ansari 2019b). The EPSNN was determined to be a better model for exchange rate prediction than a backpropagation neural network (BPNN) and other models (Sahu et al. 2019). To determine the uncertainty associated with stock data, Nayak et al. proposed ACFLN and determined that it is more efficient in handling uncertainty than the MLP, BPNN, auto-regressive integrated moving average, and other models (Nayak et al. 2018). Based on the literature, HONNs and their variants (hybridized HONNs and metaheuristic-based learning) are widely applied in FTS, particularly in the domains of stock index prediction, bitcoin price prediction, wind energy forecasting, function regularization, classification, and electricity consumption. Evidently, HONNs and their variants have achieved distinguished performance in terms of a high accuracy,

fast convergence, and fewer prediction errors in FTS forecasting. Although several evolutionary algorithms have been used for HONN training to solve intricate problems, their efficiency is hindered by the learning parameters.

Many applications of AEFA have been reported in the literature. Table 1 lists some recent engineering applications of AEFA and its improved versions.

Behera et al. applied the basic AEFA to optimize an ANN structure for software reliability dataset forecasting (Behera et al. 2022). The ANN trained with AEFA yielded a better model than that obtained using other algorithms. In 2021, Al-Khraisat et al. used the AEFA for the optimum placement of PMU (Al-Khraisat, AL-Dmour and Al-Maitah 2021). Nayak et al. incorporated the concept of elitism into the basic AEFA to train a neuro-fuzzy model for the compressive strength prediction of concrete structures and established the competitiveness of the AEFA (Nayak et al. 2021). An improvement in the AEFA through the inclusion of an inertia factor and the concept of repulsive force was claimed by Bi et al. in 2022. They coined the IRAEFA model and applied it to the spherical mining spanning-tree problem. Their improved version was determined to be more effective than the other metaheuristics. However, this is associated with long execution times. To supplement the convergence and poor search capability of the AEFA, a scheme for generating Coulomb's constant was proposed in Cheng et al. (2022). A log-sigmoid function was suggested for generating the Coulomb's constant instead of an exponential function, which decreases rapidly through evolution. The improved version was tested with 18 benchmarked functions as well as the optimization of neural networks. The Nelder-Mead simplex algorithm integrated with AEFA for the optimization problem was proposed in Izci et al. (2020). The former method helps achieve an improved local search ability, and the latter helps achieve a global search capacity. Another improvement found by the authors in Houssein et al. (2021) incorporated strategies such as a modified local escaping operator, opposition-based learning, and levy flight with the basic AEFA. The new version was tested on the parameter optimization of CEC's 2020 functions and fuel cell; it was determined to be superior to nine other metaheuristics. The AEFA potential in determining the controlling parameters of an automatic voltage regulator was established in Demirören et al. (2019). The AEFA accurately estimated the undefined parameters of the triple-diode model of a photovoltaic unit in Selem et al. (2021). The AEFA pattern search method was adopted in Alanazi and Alanazi (2022) for distribution network reconfiguration. The results highlight the improved performance of the anticipated technique in achieving a lower value of different objectives than the conventional AEFA, PSO, and grey wolf optimization (GWO) methods based on many-criteria reconfiguration. Zheng et al. proposed a sin-cosine-based AEFA for logistic distribution vehicle routing (Zheng et al. 2022). The sine-cosine update mechanism was integrated with the AEFA, which helps achieve dynamic steadiness between the global and local searches of the AEFA. The basic AEFA was hybridized with a cuckoo search algorithm and refractive learning in Adegboye and Deniz Ülker (2023). The hybrid model was evaluated using benchmark function optimization and achieved a better convergence speed and search ability than the basic AEFA. The aforementioned studies show rapid growth in the application of the AEFA and its variants to engineering optimization. However, AEFA applications in the domain of data mining lack, specifically in FTS forecasting, requiring further exploration.

Table 1 AEFA applications and variants in engineering optimization

Author(s) [Ref.]	Year of publication	Applied methods	Application area	Major findings of the study
Yadav et al. (2019)	2019	Novel AEFA	Benchmark function optimization	AEFA has established as a novel and efficient optimization algorithm
Anita et al. (2020)	2020	AEFA with CSS, MOA, PSO and GSA	Stability condition checking of AEFA	AEFA works fine for stability condition checking
Anita et al. (2019)	2019	AEFA compared with PSO, GA, ABC, and GSA	AEFA is tested for two bench mark problem that is six and fifteen generator power plant systems	Convergence rate is fast in case of AEFA
Behera et al. (2022)	2022	AEFA + ANN	AEFA + ANN model used to predict software reliability datasets	Proposed model is best suitable for forecasting of software reliability datasets
Al- Khraisat et al. (2021)	2021	AEFA	Placement of phasor measurement units using optimization algorithm	AEFA is best suitable for OPP problem
Nayak et al. (2021)	2021	Elitism AEFA based Neuro-fuzzy predictor	Estimation of compressive strength of concrete cements	AEFA found effective in selecting optimal parameters of a neuro-fuzzy based predictor
Bi et al. (2022)	2022	AEFA with inertia and repulsion (IRAEFA)	Spherical mining spanning tree problem	IRAEFA has better performance over basic AEFA and eight other metaheuristics
Cheng et al. (2022)	2022	Log sigmoid for generation of Coulomb's constant in AEFA	18 benchmark function and ANN optimization problem	The improved version found effective in achieving a faster and efficient AEFA
Izci et al. (2020)	2020	Nelder-Mead simplex method with AEFA	Four benchmark functions optimization	Better performance capability than basic AEFA and other comparative methods
Houssein et al. (2021)	2021	AEFA with opposition-based learning, levy flight and modified local escaping operator	CEC'2020 function and parameter optimization of fuel cell	Proposed method is better than 9 well known metaheuristics. These results prove the superiority of proposed method compared to other methods in determining the unknown parameters of the PEMFC model
Demirören et al. (2019)	2019	AEFA	Determining controller parameter in automatic voltage regulator system	AEFA has good potential due to its statistical data and transient response
Salem et al. (2021)	2021	AEFA	Parameter extraction of three-diode photovoltaic model	AEFA achieved accurate estimation of all undefined parameters of the triple-diode model of a photovoltaic unit
Alanazi and Alanazi (2022)	2021	AEFA pattern search method based on many-criteria optimization	Distribution networks reconfiguration	AEFA's shown better performance compared to previous studies in solving reconfiguration problem
Zheng et al. (2022)	2022	Enhanced AEFA with Sine-Cosine mechanism	Logistic distribution vehicle routing	Improve the scheduling ability and efficiency of logistics distribution vehicles
Adegboye and Ülker (2023)	2023	Hybrid AEFA with Cuckoo search algorithm with refraction learning	Benchmark function optimization	Advance convergence speed and solution precision. Cuckoo search boosted the convergence and exploration of AEFA where refraction learning enhanced the exploration of basic AEFA

Several schemes have been proposed to modify existing metaheuristic algorithms and improve their accuracy. OBL is a popular concept introduced in 2005 to enhance the performance of machine learning algorithms (Tizhoosh 2005). Subsequently, several attempts have been made to establish theoretical extensions, incorporating OBL with existing metaheuristics to improve their performance as well as real-world engineering applications. A systematic survey was conducted in Mahdavi et al. (2018) that summarized the use and growth of OBL. Table 2 summarizes recent advances in OBL and its integration with existing learning algorithms.

The mathematical and theoretical aspects of OBL were determined by Tizhoosh in 2005. A compressive survey of OBL, its variants, and real-world engineering applications was conducted (Mahdavi et al. 2018). An improved crow search algorithm (ICSA) with OBL was proposed in Shekhawat and Saxena (2020), revealing the competitive performance of ICSA over other methods. To enhance the exploration capacity of the whale optimization algorithm, OBL was integrated and determined to be better than competitive models in evaluating different benchmarked functions as well as in estimating the parameters of solar cell diode models (Abd Elaziz and Oliva 2018). The same author proposed an enhanced sine–cosine algorithm with OBL for global optimization, claiming a fast convergence (Abd Elaziz et al. 2017). In 2020, Tubishat et al. used OBL and a local search algorithm to increase the population diversity and exploration capacity of the salp swarm algorithm and applied this method on 18 benchmark datasets from the UCI repository for the feature selection problem (Tubishat et al. 2020). The economic dispatch problem of a power system was solved in Pradhan et al. (2018) using a grey wolf optimization algorithm integrating OBL (OGWO). The proposed method accelerated the convergence rate of the standalone method in terms of computational and fuel costs. Jain and Saxena proposed OBL moth flame optimization (OB-MFO) for solving CEC-2017 functions and energy market datasets, and their study concluded with the potential effectiveness of the OB-MFO model (Jain and Saxena 2019). An opposition-based AEFA was designed by Demirören et al. (2021) for an FOPID controller design. The proposed method was determined to be statistically and computationally superior to the comparison methods. OBL has also been used with atom search optimization (which) to estimate the control parameters of an automatic voltage regulator system (Ekinici et al. 2020). For an appropriate feature selection problem, Ibrahim et al. (2019) applied OBL to social spider optimization and tested its performance. The dragonfly algorithm was coupled with OBL and applied to image segmentation (Bao et al. 2019). A genetic algorithm using OBL for FTS forecasting was developed in Kar et al. (2016). The proposed OBGA-trained ANN was determined to be superior to the ANN trained using the conventional GA. Dash et al. (Dash et al. 2021b) proposed another OBL application to predict cryptocurrency prices. The method was determined to be better than other classical predictors in terms of generating a lower prediction accuracy. A memetic search method using an opposition-based concept (OBMA) was developed to solve the maximum diversity problem (Zhou et al. 2017). The OBMA ties the best-known outcomes in most instances. The experimental analysis confirmed the effectiveness of integrating OBL with a memetic search, which significantly affected the search ability of the standard memetic search.

Table 2 OBL integrated with metaheuristics and their applications to the engineering optimization problem

Author(s) [Ref.]	Year of publication	Learning method	Application area	Major findings of the study
Tizhoosh (2005)	2005	OBL for ML	Basic concept of OBL	OBL found influential while included in existing ML and improved ANN learning
Mahdavi et al. (2018)	2018	Survey on OBL	Surveyed on different applications of OBL	A rigorous survey of 380 research articles showed that OBL can potentially be used with soft computing techniques
Shekhawat and Saxena (2020)	2020	ICSA, improved Crow search algorithm with OBL	Two benchmarks set of functions, CEC-2017, structural design, frequency wave synthesis and Model Order Reduction problems	The improved version exhibited competitive performance over contemporary optimization methods
Elaziz and Oliva (2018)	2018	OBL for improvement of search space of whale optimization algorithm	Different benchmark functions and parameter estimation of solar cells diode model	OBL helped in improving the exploration capability of WOA and performed better than other competitive models
Elaziz and Oliva (2017)	2017	Improved Sine–Cosine algorithm with OBL	Solving benchmark functions	OBL amended the search ability of SCA. The proposed method achieved better convergence and accuracy
Tubishat et al. (2020)	2020	Improved Salp swarm algorithm with OBL and Local search algorithm	18 datasets from UCI for feature selection	OBL helps in population diversity of Salp Swarm algo. And Local search algo. Improved the exploration capacity of Salp Swarm algo
Pradhan et al. (2018)	2018	OGWO-opposition based grey wolf optimization	Economic dispatch problem	OBL accelerated the convergence of grey wolf optimization, reduced the computational cost
Jain and Saxena (2019)	2019	OB-MFO: oppositional concept-based moth flame optimization	CEC-2017 benchmark functions, volatility and market power analysis of energy data	Effectiveness of OB-MFO has been suggested. Proposed method is more profitable and can be a potential tool
Demirören et al. (2021)	2021	ObAEF – oppositional concept based AEF	FOPID controller design for unstable magnetic ball suspension system	ObAEF method performed better to other computationally and statistically
Ekinci et al. (2020)	2020	OBL with atom search optimization (OBASO)	Estimating parameters of automatic voltage regulator system	Superior control performance of OBASO over classical methods in estimating AVR parameters
Ibrahim et al. (2019)	2019	OBL with social spider optimization	Feature selection and classification problem	Effective selection of relevant features and improved the search ability of base optimizer
Bao et al. (2019)	2019	OBL with Dragonfly algorithm (OBLDA)	Multilevel thresholding color image segmentation	Proposed method got high accuracy and stability

Table 2 (continued)

Author(s) [Ref.]	Year of publication	Learning method	Application area	Major findings of the study
Kar et al. (2016)	2016	OBL with GA	Financial time series forecasting	OBGA trained ANN produced lower MAPE than standalone GA trained ANN
Dash et al. (2021b)	2021	Quasi opposition-based Rao algorithm for ANN training (QORA-ANN)	Cryptocurrency prices prediction	Proposed model produced reduced prediction error compared to traditional optimization of ANN
Zhou et al. (2017)	2017	Opposition based memetic search (OBMA)	80 large benchmarks with 2000–5000 instances from the literature on maximum diversity problem	OBMA established suitable balance between exploration and exploitation and found improved best solutions
Yildiz et al., (2022)	2022	OBL with grasshopper optimization algorithm	Engineering optimization problems	The OBL helped in enhancing performance of grasshopper optimization algorithm

Elitism helps retain healthy individuals across generations in evolutionary algorithms. An elite opposition-based learning methodology was used to advance the Grasshopper optimization algorithm (Yildiz et al. 2022). The improved method was applied to solve several engineering problems and determined to be effective. A distance strategy-based elitism method was proposed for selection operations in evolutionary algorithms (Du et al. 2018). In addition, elitism was used with the GA to facilitate the layout design (Jerin Leno et al. 2016). The sine cosine algorithm (SCA) with an elitism strategy was used in Sindhu et al. (2017) to select discriminative feature sets that enhanced the classification accuracy of high-dimensional datasets. A chaotic map was integrated into the Henry gas solubility optimization algorithm to enhance the convergence rate, and the robustness of the resulting method was tested and established to solve various constraint optimization problems in the areas of manufacturing and mechanical design (Yildiz et al. 2022a). The grasshopper optimization algorithm has been hybridized with the Nelder-Mead algorithm and performed well in real-world engineering optimization problems, such as in designing robot grippers (Yildiz et al. 2021). The Nelder-Mead algorithm was hybridized with the salp swarm optimization method for the structural optimization of electric vehicle components (Yildiz 2020). In incorporating chaotic maps with a levy flight distribution, a new hybrid algorithm for engineering optimization was suggested in Yildiz et al. (2022b). RNNs, such as long short-term memory (LSTM), have been suggested for modelling sequence data. As FTS are sequential, these methods can be used as effective approximations of such data. Bai et al. systematically evaluated generic recurrent networks, such as LSTMs and convolutional architectures, in sequence modelling using a broad range of datasets across diversified tasks (Bai et al. 2018). The results suggest the superior performance of a simple convolutional architecture over LSTM, i.e., a temporal convolutional network. An attention mechanism-based sequence transduction model, called the transformer, was proposed to replace recurrent layers in the encoder-decoder model (Vaswani et al. 2017). This approach uses scaled dot-product attention and multi-headed self-attention methods to induce global input–output dependencies. A transformer assumedly trains significantly faster than models based on recurrent and convolutional layers.

A FLANN produces higher-order effects of input signals through nonlinear functional transformations via links. The attributes of an input pattern were expanded into several terms and passed through a functional expansion unit. The *sine* and *cosine* trigonometric functions are used to expand the original input dimensions. For example, input x_i expands into several terms through trigonometric expansion functions such as $c_1(x_i) = (x_i)$, $c_2(x_i) = \sin(x_i)$, $c_3(x_i) = \cos(x_i)$, $c_4(x_i) = \sin(\pi x_i)$, $c_5(x_i) = \cos(\pi x_i)$, $c_6(x_i) = \sin(2\pi x_i)$, and $c_7(x_i) = \cos(2\pi x_i)$. The weighted sum of the functional expansion unit outputs is passed to the activation function to estimate the value of the output neuron. The output of the model is then compared with the target output, and the absolute deviation, called the error signal, is calculated. The accumulated error signal is propagated back to train the model.

For a given input pattern, the FLANN computes an output as follows. Let $X(n) = \{x_i, x_{i+1}, \dots, x_n\}$, where ($n = \text{input vector size}$), be an input vector. Using trigonometric basis functions, this vector is expanded nonlinearly as $X_{\text{expanded}}(N)$. Given the input $X_{\text{expanded}}(N)$, the model produces an output $\hat{y}(n)$ as shown in Eq. (1):

$$\hat{y}(n) = X_{expanded}(N) \times W(n) + bias, \tag{1}$$

where $W(n)$ are the weight associated with the n^{th} pattern. This output is then passed through a sigmoid activation to produce an output $y(n)$ as shown in Eq. 2. The $error(n)$ is calculated using Eq. (3).

$$y(n) = \frac{1}{1 + e^{-\hat{y}(n)}} \tag{2}$$

$$error(n) = d(n) - y(n) \tag{3}$$

The weights are updated as shown in Eq. (4 – 5) using adaptive learning rules:

$$w_{ij}(t + 1) = w_{ij}(t) + \mu * \Delta(t), \tag{4}$$

$$\Delta(t) = \delta(t) * [f(x_i)], \tag{5}$$

where $(t) = [\delta_1(t), \delta_2(t), \dots, \delta_k(t)]$, $\delta_i(t) = (1 - \hat{y}_i^2(t) * e_i(t))$, and μ is the learning parameter.

The PSNN computational model is as follows. It has a two-layered, fully connected feed-forward network architecture. The first layer comprises sigma units (summing), and the second layer is the product (pi) layer. The inputs are connected to the neurons of the sigma layer, and their outputs are fed to the neurons in the pi layer. The weights and thresholds of the input-summing layer are trainable, whereas those of the summing-product layer are set to unity. Owing to the single adjustable parameter set, the training time was drastically reduced. Neurons at the summing units use linear activations and those at the product units use nonlinear activations of the network output. The product units provide a higher-order capability by expanding the lower-dimensional input space into higher dimensions. The higher-dimensional space achieved offers a superior nonlinear separability without exponential growth in the weights. The output of the j th sigma unit is computed using the weight sum of each input x_i and corresponding weight w_{ij} as in Eq. (6):

$$y_j = \sum_{i=1}^n w_{ij} * x_i, \tag{6}$$

where n denotes the input size. The pi-layer neuron computes the product of the outputs of the sigma units and applies a nonlinear activation to it, as in Eq. (7):

$$y = \sigma \left(\prod_{j=1}^k y_j \right), \tag{7}$$

where k is the number of sigma units, and is the order of the network. The error signal $error(n)$ was calculated as follows:

$$error(n) = d(n) - y(n). \tag{8}$$

We reviewed recent articles on financial forecasting, different HONN applications for FTS forecasting, different metaheuristics for HONN optimization, OBL integration with existing metaheuristics, and the elitism concept in improving the learning capacity. We obtained the following insights from these studies:

- Automotive predictive systems with improved accuracies remain lacking in the large-scale FTS forecasting domain.
- Metaheuristic-based HONNs have demonstrated promising performances in different data-driven problems, and their proficiency in FTS forecasting must be assessed in depth.
- The AEFA has emerged as a competitive optimization mechanism. However, AEFA-based HONNs for FTS forecasting remains limited.
- The convergence rate and population diversity of the basic AEFA must be improved to maintain a balance between its exploration and exploitation capacities.
- Finally, no predictive framework has been established that integrates the improved AEFA and HONN for large-scale FTS forecasting.

Proposed EOAEFA + HONN forecasting method

This section describes the proposed EOAEFA + HONN based forecasting method in two phases; Phase 1: improved the AEFA by incorporating elitism and OBL into the basic AEFA, called EOAEFA, and Phase 2: search for the optimal parameters of the HONN using EOAEFA.

Design of EOAEFA

As the basis of the proposed EOAEFA is the AEFA, OBL, and concept of elitism, recalling may be helpful to understand the working principles of the EOAEFA.

Basis of EOAEFA

The AEFA mimics a charged particle as an agent in the search space, and the strength of such an agent can be measured in terms of its charge. A mass of these charged particles floats in the search domain with the help of electrostatic attraction and repulsive forces. Particles can interact with each other through their charges. The positions of these charges are measured as potential solutions to the target problem. The force of attraction is considered only in the basic AEFA, meaning that all particles associated with lower charges are attracted to the particle with the highest charge, called the best particle or individual. The position of the i^{th} charged particle (X_i) at time t is represented by Eq. (9):

$$X_i(t) = (X_i^1, X_i^2, X_i^3, \dots, X_i^D), i = 1, 2, 3, \dots, N \text{ and } d = 1, 2, 3, \dots, D, \quad (9)$$

where N and D are the total numbers of charged particles and parameters (dimensions), respectively. The position of the i^{th} particle at time $(t + 1)$ is updated as in Eq. (10) when it achieves the best fitness value.

$$P_i^d(t + 1) = \begin{cases} X_i^d(t + 1) & \text{if } \text{fitness}(X_i(t + 1)) \leq \text{fitness}(P_i(t)) \\ P_i^d(t) & \text{if } \text{fitness}(X_i(t + 1)) > \text{fitness}(P_i(t)) \end{cases} \quad (10)$$

The charge associated with the i^{th} particle ($Q_i(t)$) at time t is expressed as Eq. (11):

$$Q_i(t) = \frac{q_i(t)}{\sum_{i=1}^N q_i(t)} \quad i = 1, 2, \dots, N, \tag{11}$$

where $q_i(t)$ is a suitable charge function calculated as in Eq. (11) using the best- and worst-fit particles in the search space.

$$q_i(t) = \exp\left(\frac{fitness_i(t) - fitness_{worst}(t)}{fitness_{best}(t) - fitness_{worst}(t)}\right) \tag{12}$$

The force $F_{ij}^d(t)$ experienced at the i^{th} particle holding charge $Q_i(t)$ because of the j^{th} particle holding charge $Q_j(t)$ is defined as follows:

$$F_{ij}^d(t) = K(t) \frac{Q_i(t) \cdot Q_j(t) \cdot (P_j^d(t) - X_i^d(t))}{X_i(t) - X_j(t)^2 + \varepsilon}, \tag{13}$$

where $K(t)$ is Coulomb's constant calculated in terms of the current and maximum iteration (as in Eq. (14)) and ε is a small positive constant.

$$K(t) = K_0 \cdot \exp\left(-\alpha \frac{iteration}{max.iteration}\right) \tag{14}$$

The value of parameter $\alpha = 30$ and $K_0 = 500$. A larger initial value of K_0 helps in exploring the search process and gradually decreases through the iterations to regulate the accuracy. The resultant electrostatic force F_i^d acting on the i th particle at time t can be calculated as in Eq. (15) and the electric field is calculated as in Eq. (16).

$$F_i^d(t) = \sum rand \cdot F_{ij}^d(t), j = 1, 2, \dots, N, \text{Nandi} \neq j \tag{15}$$

$$E_i^d(t) = \frac{F_i^d(t)}{Q_i(t)} \tag{16}$$

Per Newton's law of motion, the acceleration $a_i^d(t)$ of the i th charged particle with unit mass $M_i(t)$ at time t is computed as in Eq. (17).

$$a_i^d(t) = \frac{Q_i(t) \cdot E_i^d(t)}{M_i(t)} \tag{17}$$

The velocity and position of the i th charged particle at time $(t + 1)$ are updated according to Eqs. (18) and Eq. (19), respectively.

$$V_i^d(t + 1) = rand_i * V_i^d(t) + acceleration_i^d(t) \tag{18}$$

$$X_i^d(t + 1) = X_i^d(t) + V_i^d(t + 1) \tag{19}$$

The particle associated with the maximum quantity of charge can be considered the best individual. This individual particle attracts other particles with a lower charge and fewer voyages in the search domain.

Population-based search methods begin their exploration process with the initialization of a collection of potential candidate solutions called the initial population. The initialization of a population with random points in the search space is a common method. Exploration begins with these random solutions and is directed towards the global optima with the application of several control parameters and mechanisms. Occasionally, these random initial points lead the optimization algorithm towards a local optimum. To avoid such a trap, the concept of OBL, which is based on the theory of an opposite point, has been proposed. According to OBL, both the initial population and its opposite population are included in the search space, which considers the existence of possible solutions in any direction, thereby enabling the algorithm to effectively explore the search space. In addition, an opposite number \bar{x} of a given real number $x \in [lb, ub]$ in one dimension can be computed as in Eq. (20):

$$\bar{x} = lb + ub - x, \quad (20)$$

where lb and ub denote the lower and upper bounds of the search domain, respectively. Equation (12) can be extended to a multidimensional space. Let $x \in R^n$, $x = [x_1, x_2, \dots, x_n]$, where $x_i \in R$; the opposite point $\bar{x} = [\bar{x}_1, \bar{x}_2, \dots, \bar{x}_n]$ can be computed as in Eq. (21).

$$\bar{x}_i = lb_i + ub_i - x_i, i = 1, 2, \dots, n \quad (21)$$

In OBL, if the fitness function value of position x is inferior to that of its opposite position \bar{x} (i.e., $fitness(x) < fitness(\bar{x})$), it is replaced by \bar{x} ; otherwise, x is saved. The population is updated with the better value of x or \bar{x} .

Maintaining steadiness between intensification and diversification in an evolutionary search algorithm is a critical factor that significantly affects its performance. Elitism is applied to the selection operation of evolutionary algorithms for this purpose. This strategy retains the most beneficial candidates and benefits exploitation. In practice, a small value is often set for the degree of elitism strategy or number of elites. The elitism process includes a few best individuals from one generation in the population of the following generation. The key purpose of using elitism is to preserve the positions of promising parts of the search space across generations and enable the continuous exploitation of these promising areas. Furthermore, it ensures the existence of the best individuals by considering the entire processing of an algorithm in the last generation created, which is the final outcome.

EOAEFA algorithm

The deprived convergence rate and trapping in a local best optimum are the two weaknesses of the basic AEFA that influence its overall performance. It updates the current particles in the search space towards the local best solution obtained thus far and may ignore some better-fitting solutions that are in opposite directions from the current particle. The EOAEFA method avoids simultaneous consideration of a solution and its opposite. This helps improve the exploration ability of the basic AEFA. In addition, in each iteration, it saves the highly fit solutions as elites and carries them over to the next iteration, thereby helping achieve a better convergence. The EOAEFA does not significantly

affect the configuration of the basic AEFA but improves its accuracy through the inclusion of OBL and elitism. Algorithm 1 presents the steps of the EOAEFA.

Algorithm 1: EOAEFA

1. */*Initialization*/*
 2. Random initialization of population P of N individuals in the search range within the operating limits, $\alpha = 30$, $K_0 = 500$, M , and $MaxIter$
 3. */*Generate opposite population \bar{P} */*
 1. *for* $j = 1:N$
 2. *for* $i = 1$: number of design variable
 3. $\bar{P}(j, i) = P(\cdot, i)_{min} + P(\cdot, i)_{max} - P(j, i)$
 4. *end*
 5. *end*
 4. */*Evaluation*/*
 5. Measure the fitness of each particle in P and \bar{P}
 6. Sort P and \bar{P} based on the fitness values
 7. Select N fittest solutions from $P = P \cup \bar{P}$
 8. */*Reproduction and update*/*
 9. $Iter = 1$.
 10. **While** ($Iter <= MaxIter$)
 11. Calculate K , $bestfit$, and $worstfit$
 12. *for* $i = 1:N$ *do*
 13. Evaluate the fitness values of the i^{th} particle
 14. Calculate total force $F_i^d = \sum rand \cdot F_{ij}^d, j = 1, 2, \dots, N$ and $i \neq j$
 15. Calculate acceleration $a_i^d = \frac{Q_i * E_i^d}{M_i}$
 16. Update velocity $V_i^d = rand_i * V_i^d + a_i^d$
 17. Update position $X_i^d = X_i^d + V_i^d$
 18. *end for*
 19. */*Apply elitism*/*
 20. Identify M elite solutions and the worst solutions of P
 21. Replace the worst solutions with elite solutions and update P
 22. $Iter = Iter + 1$
 23. **end while**
 24. Return **best** solution.
-

The EOAEFA method begins with the initialization of the algorithm-specific parameters and a randomly initialized population P . The oppositional population \bar{P} is then generated from P . Both P and \bar{P} are evaluated simultaneously, and N best-fit particles are selected from $P \cup \bar{P}$. Next, the AEFA operators are applied to the updated population to intensify and diversify the search space. The positions and velocities of all particles were updated through the iterations. At the end of each iteration, M elite particles are identified. The value of M must be carefully selected and assigned a small number. A larger value of M may significantly alter the quality of the solutions, and many duplicate solutions can exist. Finally, the same number of worst solutions was replaced by these elite solutions. Hence, the elite particles are carried to the next iteration, retaining the finest solutions obtained thus far and advancing the convergence rate of the algorithm.

EOAEFA + HONN-based forecasting

The proposed EOAEFA was used to adjust the parameters of the two HONNs: PSNN and FLANN; hence, two hybrid models were formed: EOAEFA-PSNN and EOAEFA-FLN. This process is explained as follows. An arbitrary HONN structure can be mapped onto a particle or agent of the EOAEFA. Therefore, the EOAEFA population can be

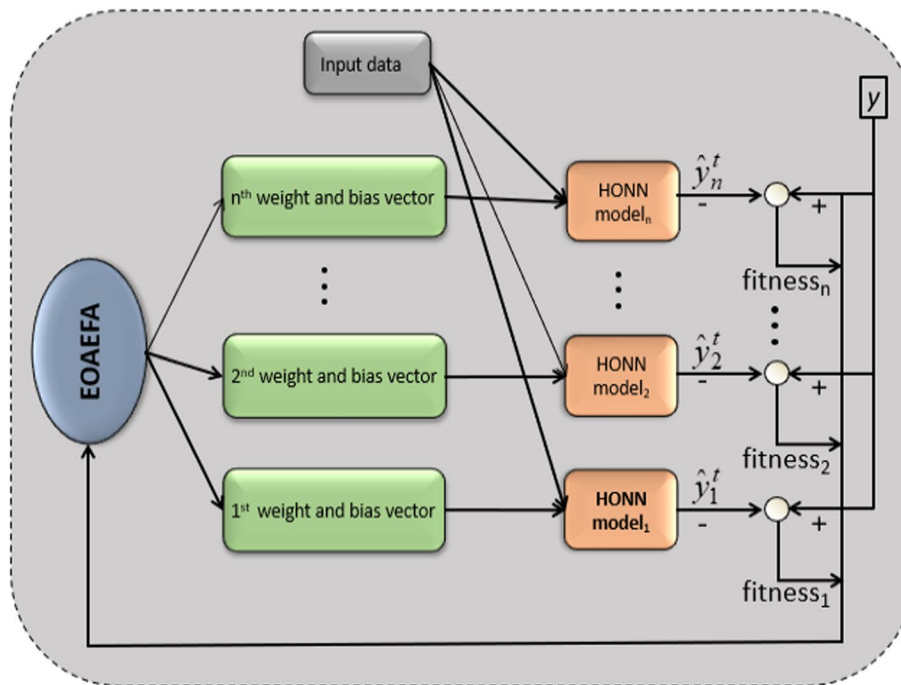


Fig. 1 EOA-EFA + HONN based forecasting

viewed as a set of potential HONN structures. Each HONN structure, along with the training data, was evaluated and the corresponding fitness values were calculated as in Eq. (22).

$$fitness(HONN_i) = \frac{1}{\sum_{i=1}^{inputsize} |predicted_i - actual_i| / inputsize} \tag{22}$$

The EOA-EFA is then applied to execute search operations as explained in the previous section on a population of the HONNs. The particles compete, and finally, the best solution, i.e., the best HONN architecture, evolves. The test data are then fed to the best HONN, and the deviation of the HONN output from the actual output is extracted and considered as the performance of the corresponding HONN model; the lower the deviation, the better the fitness of the HONN. Figure 1 illustrates this process.

FTS data and statistical analysis

For experimental purposes, real FTS datasets were collected from Internet sources. Four closing prices series, NASDAQ, DJIA, S&P 500, and Russell 2000, were downloaded from <https://finance.yahoo.com/>, considering financial transactional days from August 23, 2021 to August 19, 2022 on a daily basis. Each FTS consists of 255 index records with comprising the date, open, high, low, close, adj. close prices, and volume. We considered open, high, low, and closed price values only for the experiment. Similarly, four exchange rate series, Bitcoin (BTC), Euro (EUR), British Pound (GBP), and Japanese Yen (JPY), were collected from the same source in daily volume against the US Dollar, with each series containing 255 records. Another three FTS, crude oil

Table 3 Statistical properties of the closing prices series

Statistic	DJIA	S&P 500	NASDAQ	Russell 2000
Minimum	2.9806e + 04	2.0631e + 04	1.1773e + 03	174.9000
Mean	3.4008e + 04	2.3318e + 04	1.3772e + 04	2.0338e + 03
Median	34,367	2.3554e + 04	1.4497e + 04	2.0346e + 03
Variance	2.6599e + 06	1.0900e + 06	9.1354e + 06	7.9047e + 04
Maximum	36,675	2.5267e + 04	16,575	2.5932e + 03
Standard deviation	1.6309e + 03	1.0440e + 03	3.0225e + 03	281.1527
Skewness	- 0.6243	- 0.5313	- 2.9437	- 2.9336
Kurtosis	2.4683	2.4387	12.6564	19.8676
Correlation coefficient	0.0205	0.0721	- 0.5124	- 0.4951
Ljung-Box test	h = 1, p = 0.0000	h = 1, p = 0.0000	h = 1, p = 0.0000	h = 1, p = 0.0000
KPSS test	h = 1, p = 0.0100	h = 1, p = 0.0100	h = 1, p = 0.0100	h = 1, p = 0.0100
ADF test	h = 0, p = 0.5089	h = 0, p = 0.8077	h = 0, p = 0.0626	h = 0, p = 0.2266
PP test	h = 0, p = 0.5089	h = 0, p = 0.8077	h = 0, p = 0.0626	h = 0, p = 0.2266

Table 4 Statistical properties of the exchange rate series

Statistic	BTC/USE	EUR/USD	GBP/USD	JPY/USD
Minimum	1.9018e + 04	1.0026	1.1840	109.1490
Mean	4.0711e + 04	1.1055	1.3070	120.9415
Median	4.1821e + 04	1.1223	1.3265	115.5250
Variance	1.5540e + 08	0.0026	0.0038	86.40260
Maximum	6.7567e + 04	1.1886	1.3859	138.9670
Standard deviation	1.2466e + 04	0.0515	0.0612	9.2953
Skewness	- 0.0285	- 0.3447	- 0.5806	0.5318
Kurtosis	2.2440	1.9491	1.8736	1.7161
Correlation coefficient	0.0144	- 0.0091	0.0062	- 0.0434
Ljung-box test	h = 1, p = 0.0000	h = 1, p = 0.0000	h = 1, p = 0.0000	h = 1, p = 0.0000
KPSS test	h = 1, p = 0.0100	h = 1, p = 0.0100	h = 1, p = 0.0100	h = 1, p = 0.0100
ADF test	h = 0, p = 0.2300	h = 0, p = 0.0564	h = 0, p = 0.1063	h = 0, p = 0.9952
PP test	h = 0, p = 0.2300	h = 0, p = 0.0564	h = 0, p = 0.1063	h = 0, p = 0.9952

Table 5 Statistical properties of energy price series

Statistic	Crude oil	Natural gas	Coal price
Minimum	10.0100	1.4820	9710
Mean	64.9122	3.7350	1.1244
Median	63.8600	2.9665	11,330
Variance	646.0576	3.7836	3.7159
Maximum	123.7000	9.3290	12,606
Standard deviation	25.4177	1.9452	609.5834
Skewness	0.28360	1.0896	- 0.6352
Kurtosis	2.3024	3.3357	3.3386
Correlation coefficient	- 0.3950	- 0.1196	- 0.4043
Ljung-box test	h = 1, p = 0.0000	h = 1, p = 0.0000	h = 0, p = 0.1199
KPSS test	h = 1, p = 0.0100	h = 1, p = 0.0100	h = 1, p = 0.0293
ADF test	h = 0, p = 0.6359	h = 0, p = 0.0564	h = 0, p = 0.5043
PP test	h = 0, p = 0.6359	h = 0, p = 0.9151	h = 0, p = 0.5043

prices on daily volume, natural gas prices on daily volume, and weekly coal prices were collected from the source U.S. Energy Information Administration (website: <http://www.eia.doe.gov/>). The crude oil and natural gas price series were collected from January 2, 2020, to August 16, 2022, each containing 658 records. Fifty-seven records of weekly coal prices were available from August 8, 2021, to August 13, 2022. Different are listed in Tables 3, 4, and 5 list the statistics of the closing prices, exchange rate, and energy data series, respectively. The NASDAQ and Russell 2000 series in Table 3 show larger kurtosis values, indicating higher investment risks. All closing price series are platykurtic. The Russell 2000 series deviated significantly, whereas the other three were stable to an extent. All series showed weak correlations among their data points. The Phillips-Perron (PP) and augmented Dickey-Fuller (ADF) test results from these FTS show that the series are nonstationary. The Ljung-box test results indicate a lack of autocorrelations in the series, and the KPSS statistics support the existence of non-stationarity around a deterministic trend. Figure 2 and Fig. 3 depict the trends in closing prices and distribution of the price series, respectively.

Figure 4 shows the daily closing price data charts for the four FTS considered. The charts show frequent increases and decreases in the FTS, making the prediction of future prices difficult. Figure 5 shows the distribution of the closing price data. All four series exhibit an asymmetric distribution of data about the center, i.e., random variation. The DJIA, NASDAQ, and S&P 500 series skew left, whereas the Russell's distribution is random. The four exchange rate series and three energy prices series exhibit similar behaviors to the closing price series. Figures 4 and 5 present the exchange rate series and their distributions, respectively. Figures 6 and 7 present the energy price series and their distributions, respectively.

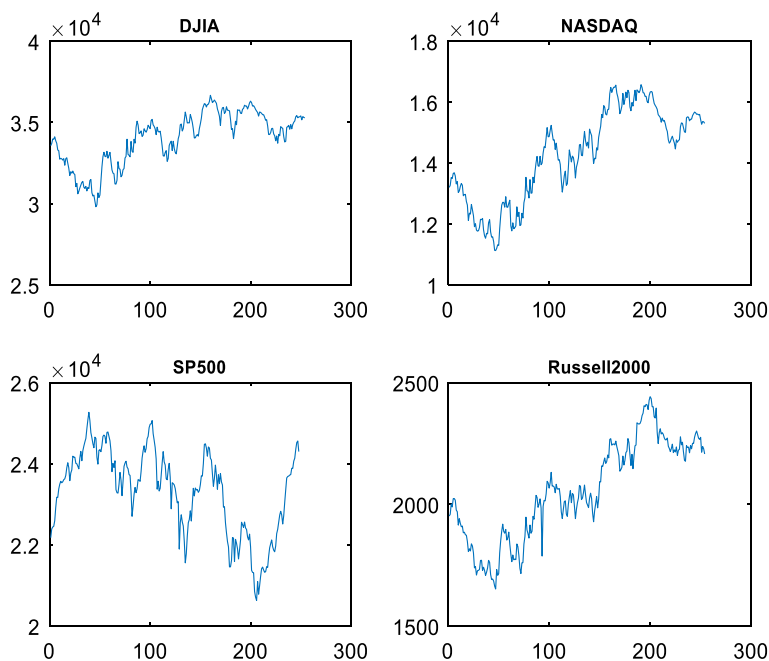


Fig. 2 Daily closing prices series

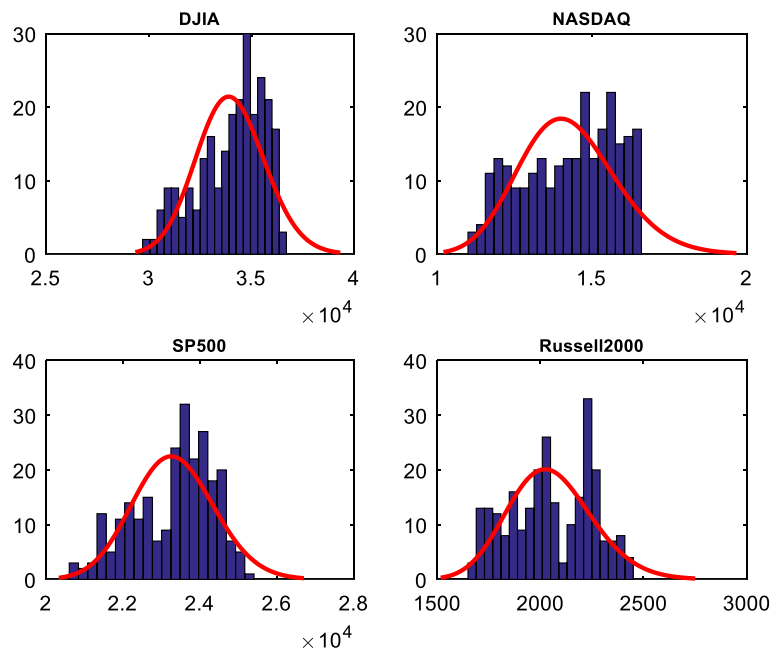


Fig. 3 Distribution of numerical closing prices of four FTS

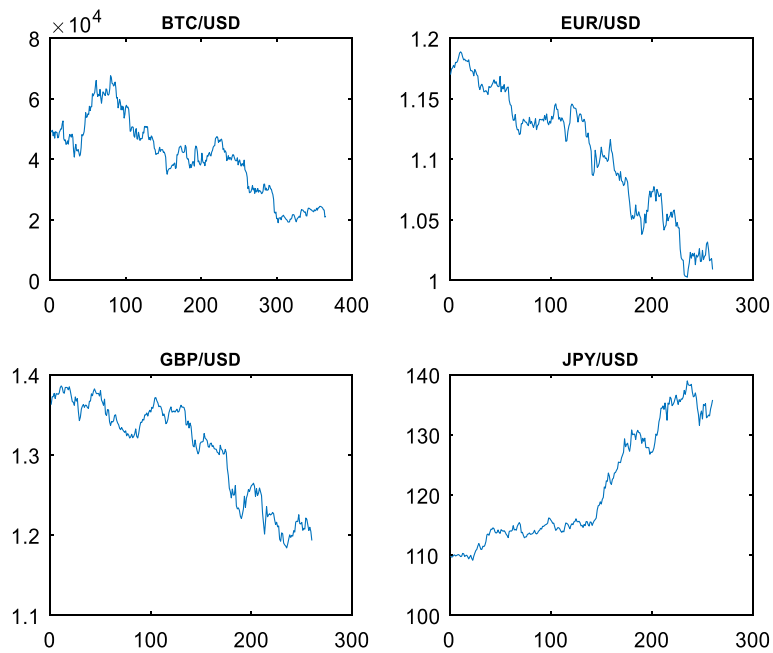


Fig. 4 Trend of numerical closing prices of four FTS

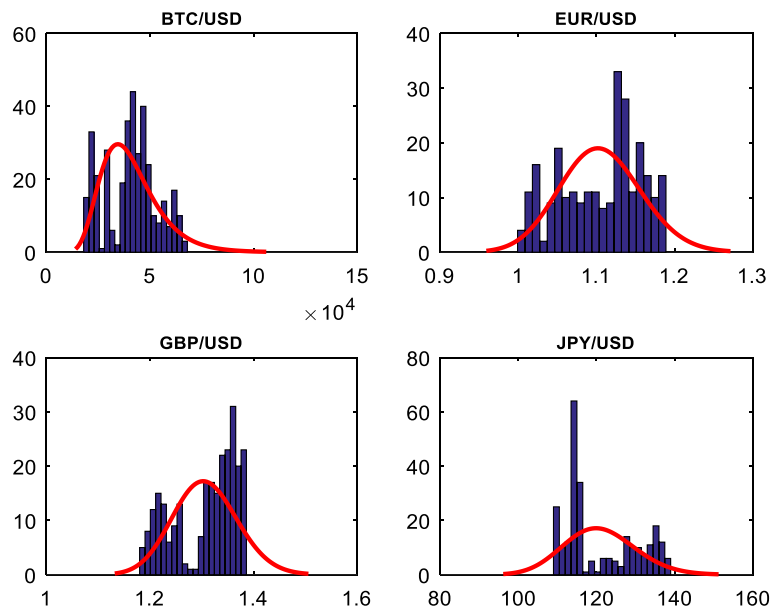


Fig. 5 Distribution of numerical closing prices of four FTS

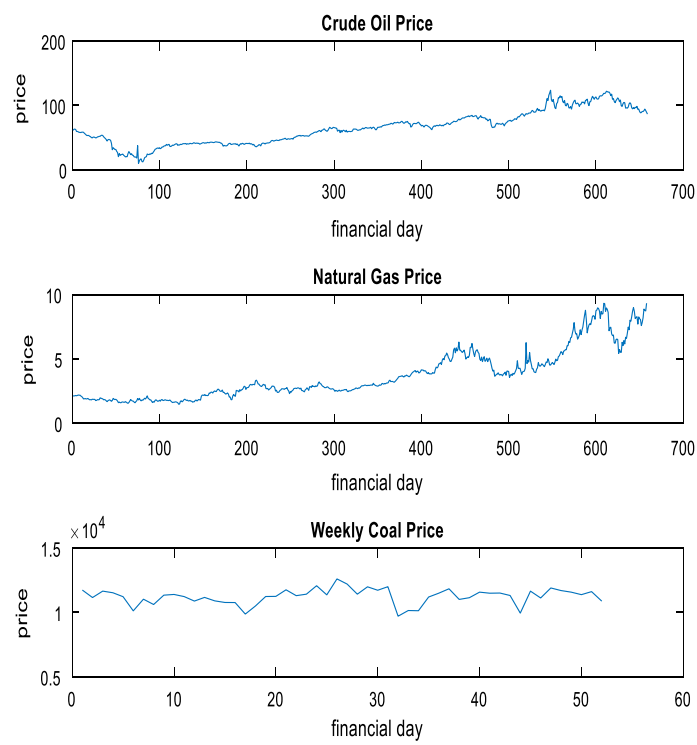


Fig. 6 Trend of numerical energy prices of four FTS

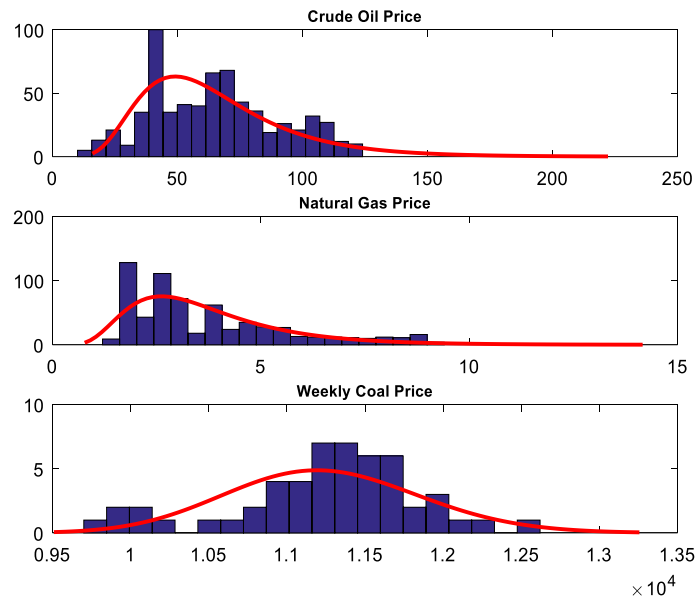


Fig. 7 Distribution of numerical energy prices of four FTS

Results and analysis

This section presents the input preparation, normalization of the model input, research design, experimental outcomes, their analysis, and comparative studies.

Model input preparation and normalization

The next step after the FTS collection and analysis is input preparation for the forecasting models and normalization of input patterns. The ordinary time-series forecasting process splits the available data into training and test sets. However, FTS

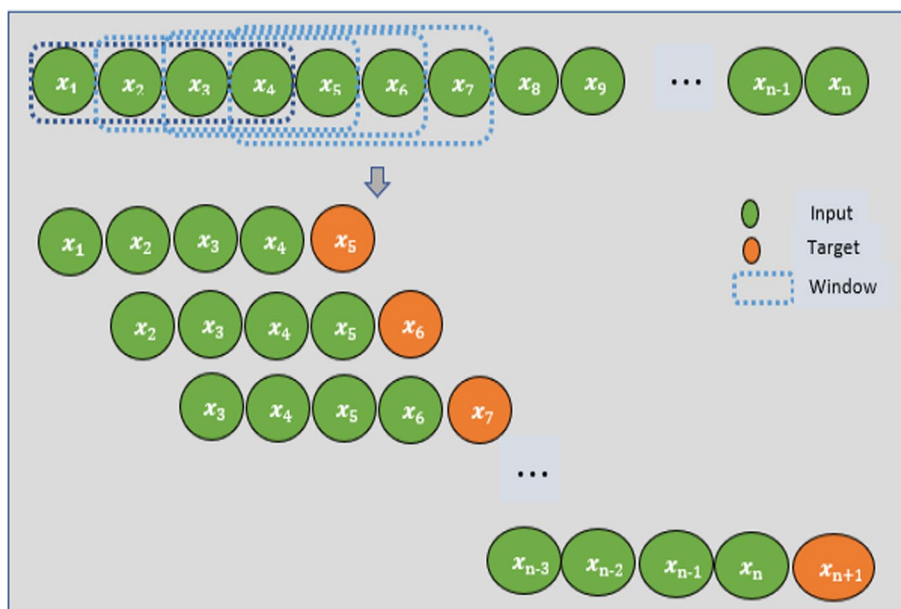


Fig. 8 Input pattern generation using sliding window process

samples cannot be selected arbitrarily or dispensed to either set, as using future values to forecast past values is nonsensical. The temporal dependence of the data must be preserved during testing. Therefore, considering current data and immediate past observations to forecast future data is important. As the FTS prediction problem is a sequence prediction task, we adopted a sliding window mechanism to generate training and test patterns from the original FTS (Nayak et al. 2014). Figure 8 provides an example of the sliding window process, where $[X_{i-k}, \dots, X_{i-2}, X_{i-1}]$ are used, and X_i is considered the target forming an input pattern. The window is then moved one step forward and the process repeats. The window size k remains fixed throughout the process and is determined experimentally. The sigmoid method, as in Eq. (23), is used to normalize data x_i into x_{norm} using the minimum (x_{min}) and maximum data points (x_{max}) of the current window under consideration.

$$x_{norm} = \frac{1}{1 + e^{-\left(\frac{x_i - x_{min}}{x_{max} - x_{min}}\right)}} \quad (23)$$

Research design

Based on the methodologies described in Sect. "Proposed EOAEFA+HONN forecasting method", different experiments were designed using standardized input patterns from 11 FTS. The window size in our experiments was selected to be 12, i.e., 12 data points from the input pattern. The same patterns were applied to all the forecasts to maintain an unbiased comparative study. Two consecutive patterns, formed by a one-step movement of the window over the series, differ only in adding new data point and dropping the oldest. Therefore, the variance in nonlinearity coupled with two consecutive patterns is nominal because we used the optimized parameters of the previous pattern for successive training instead of considering a new random parameter set. Once the network is trained using the first training set, the number of iterations for successive training sets is set to a small value. This type of adaptive training decreases the number of iterations, and thus decreases the training time. Seven models (six hybrid and one conventional ANN) were developed in a similar manner. The parameters of the two HONNs (i.e., FLANN and PSNN) were optimized using the proposed EOAEFA (forming EOAEFA-FLN and EOAEFA-PSNN-based forecasts), basic AEFA (forming AEFA-FLN and AEFA-PSNN-based forecasts), GA (forming GA-FLN- and GA-PSNN-based forecasts), and a BPNN forecast. Seven trigonometric functions (sine and cosine) were used in the FLANN for the functional expansion of the input data. Therefore, each input datum was expanded to 84 terms. For the PSNN, a 12–8–1 architecture was used as the base for the PSNN. Both types of HONNs used sigmoid activation in their neurons. The AEFA parameters were set as follows: particle size 50, $\alpha = 30$, and $K_0 = 500$ in reference to their respective articles (Yadav and Kumar 2020, 2019). The elitism factor was set to 2%, and the algorithm was iterated 100 times to reach the optimal parameter values. For the GA, the crossover and mutation probability values were set to 0.6 and 0.002, respectively. The GA was allowed 100 generations. The BPNN used an architecture of 12–25–1 neurons with a learning rate of 0.3, momentum factor of 0.4, and gradient descent learning. All experiments

were conducted using MATLAB. To compensate for the stochastic behavior of all neural forecasts, each model was executed 30 times with the aforementioned parameters, random initial weights, and threshold values. The average prediction error values from the 30 runs were recorded for performance comparisons. The mean absolute percentage error (MAPE) (Eq, (24)) was used to measure the prediction accuracy of all the forecasts.

$$MAPE = \frac{1}{No.ofpattern} \sum_{i=1}^{No.ofpattern} \frac{|Target_i - Predicted_i|}{Target_i} \times 100\% \tag{24}$$

Analysis of the results from the closing prices series

All seven forecasts were iterated 100 times, and Fig. 9 plots the error convergence graphs from the four closing price series. In all series, the EOAEFA-FLN converged the fastest except for the Russell2000 series. The convergence of the four AEFA-based models was determined to be similar to and better than that of the GA-based models. The BPNN convergence was determined to be unsatisfactory. Table 6 lists the MAPE statistics for the seven forecasting approaches from the four series. Values less than 10^{-5} are considered to be zero. The best MAPE is indicated in bold, and the second best in italics. For the NASDAQ series, the EOAEFA-FLN generated the best average error of 0.010285 whereas the EOAEFA-PSNN was second-best with an average error of 0.010823.

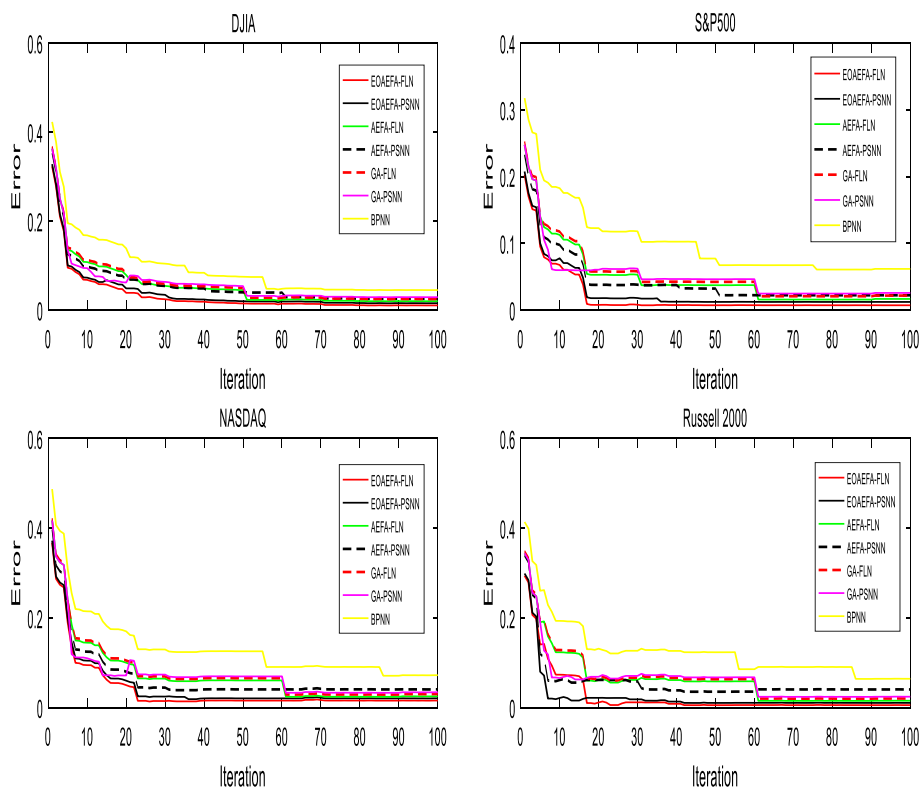


Fig. 9 MAPE convergence graphs from seven forecasts

Table 6 MAPE statistics from four closing prices series

FTS	Forecast	Minimum	Average	Maximum	Median	Standard deviation	Inter-quartile range
NASDAQ	EOAEFA-FLN	0.000000	0.010285	0.047221	0.008373	0.007950	0.009886
	EOAEFA-PSNN	0.000000	0.010823	0.047684	0.008988	0.008111	0.011372
	AEFA-FLN	0.000180	0.014485	0.046684	0.009321	0.008002	0.011099
	AEFA-PSNN	0.000000	0.015937	0.050684	0.010386	0.008518	0.012499
	GA-FLN	0.000000	0.018308	0.057221	0.011145	0.010372	0.014531
	GA-PSNN	0.000000	0.017765	0.060684	0.016854	0.010766	0.015472
	BPNN	0.000000	0.110892	0.052221	0.009024	0.008877	0.010797
DJIA	EOAEFA-FLN	0.000000	0.010895	0.053081	0.007725	0.009548	0.011486
	EOAEFA-PSNN	0.000260	0.011328	0.057081	0.008935	0.010004	0.010576
	AEFA-FLN	0.000000	0.011825	0.048081	0.009972	0.009586	0.013381
	AEFA-PSNN	0.000139	0.011111	0.051081	0.008455	0.009441	0.011531
	GA-FLN	0.000000	0.026721	0.059402	0.026534	0.012482	0.015450
	GA-PSNN	0.000284	0.023307	0.055402	0.022558	0.011706	0.015757
	BPNN	0.002081	0.050703	0.084402	0.051507	0.014441	0.015729
S&P 500	EOAEFA-FLN	0.000000	0.012230	0.059887	0.010394	0.009817	0.011509
	EOAEFA-PSNN	0.000000	0.012286	0.055887	0.011925	0.009812	0.013367
	AEFA-FLN	0.000208	0.01366	0.054887	0.01263	0.009895	0.013336
	AEFA-PSNN	0.000000	0.016202	0.051936	0.015824	0.010518	0.014362
	GA-FLN	0.000000	0.035946	0.075936	0.036803	0.014231	0.018783
	GA-PSNN	0.002366	0.041696	0.081936	0.042803	0.014711	0.018783
	BPNN	0.020113	0.081357	0.121936	0.082803	0.015643	0.018783
Russell 2000	EOAEFA-FLN	0.000000	0.009959	0.072251	0.007814	0.008571	0.010630
	EOAEFA-PSNN	0.000297	0.010636	0.068251	0.008912	0.008273	0.009840
	AEFA-FLN	0.000303	0.010884	0.067251	0.009200	0.008388	0.010089
	AEFA-PSNN	0.000104	0.013014	0.062251	0.011916	0.009258	0.012947
	GA-FLN	0.000494	0.033476	0.067071	0.034737	0.012181	0.015269
	GA-PSNN	0.001426	0.039378	0.073071	0.040651	0.012325	0.015269
	BPNN	0.032749	0.104120	0.138071	0.105651	0.013124	0.015543

Similarly, both methods retained their respective positions in the case of the DJIA and Russell FTS, and tied in the S&P 500 FTS. Overall, EOAEFA-FLN-based forecasting performs better than the other methods.

To confirm the success of the proposed forecasts, Fig. 10, 11, 12, 13 plot the model predictions against the actual closing prices for the DJIA, NASDAQ, S&P 500, and Russell 2000 FTS, respectively. Based on these plots, the nearness of the EOAEFA-HONN-based predictions to the actual closing prices is evident. The EOAEFA-FLN predictions are the closest to the actual values, followed by those of the EOAEFA-PSNN. Both models were competitive and efficient in preserving the patterns of the actual closing price series. The basic AEFA- and GA-based HONNs predictions were moderate, whereas those from the BPNN-based forecasting were poor. The stability of the proposed forecasts is confirmed by the box–whisker plots depicted in Fig. 14.

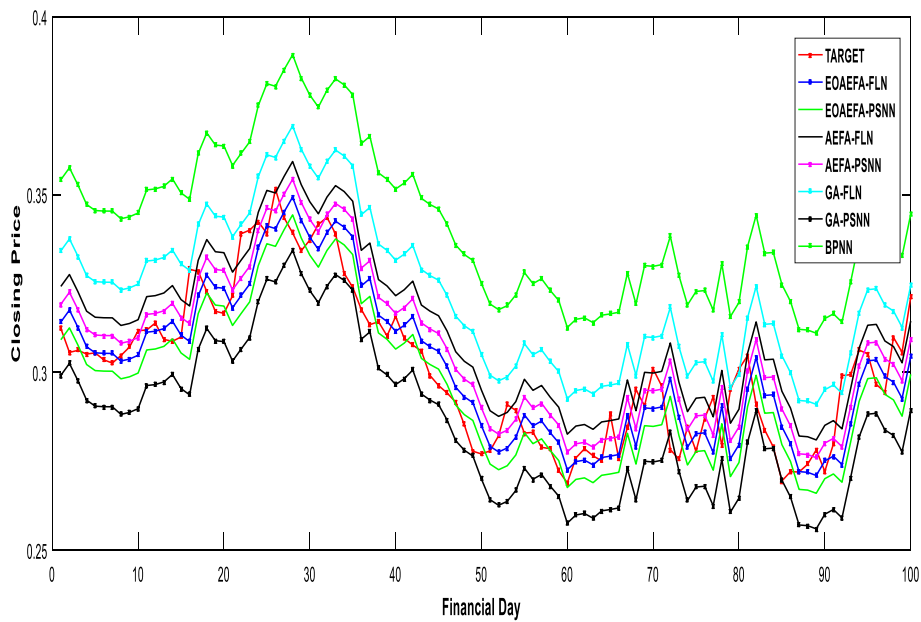


Fig. 10 Prediction plots of seven models from DJIA series

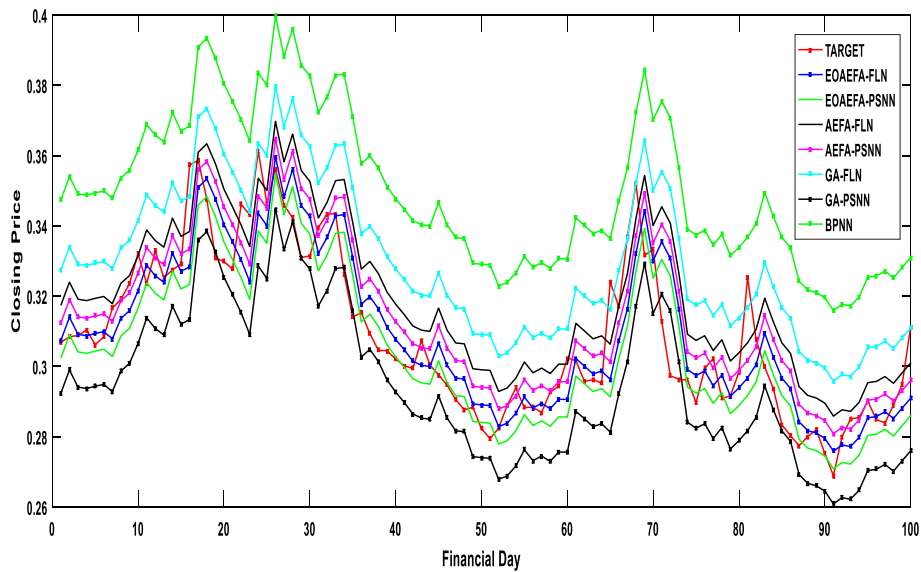


Fig. 11 Prediction plots of seven models from NASDAQ series

Analysis of results from exchange rate series

The next type of FTS data we considered for the evaluation of EOAEFA + HONN-based forecasts are four exchange rate FTS: BTC, EURO, GBP, and JPY versus US dollars. The input data preparation and normalization were the same as those in the case of closing the price series. Figure 15 visualizes the convergence rates of all forecasts. Here, the EOAEFA-FLN and EOAEFA-PSNN converged faster than the other forecasts. Table 7 summarizes the MAPE statistics. Values less than 10^{-5} are considered to be zero. The proposed forecasts achieved lower error statistics than the others;

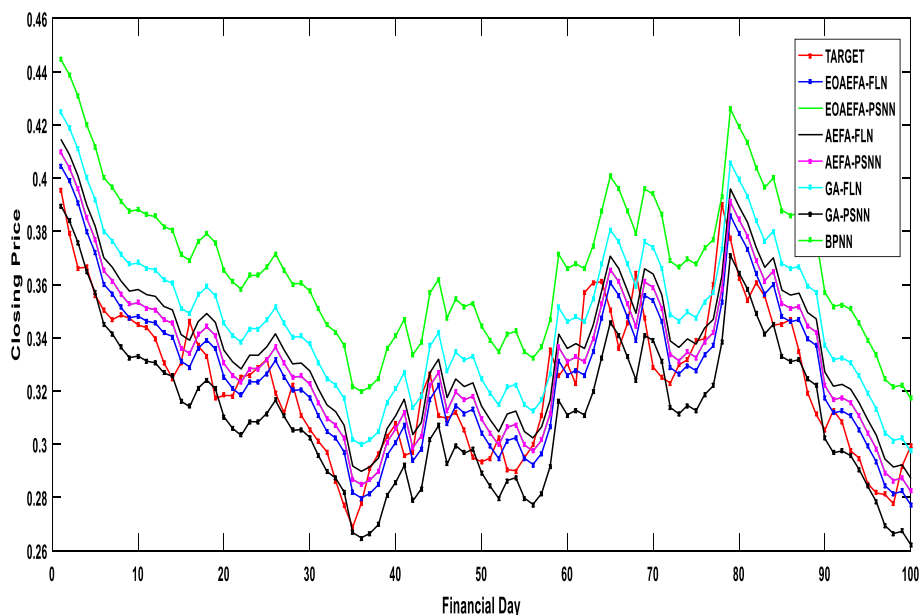


Fig. 12 Prediction plots of seven models from S&P 500 series

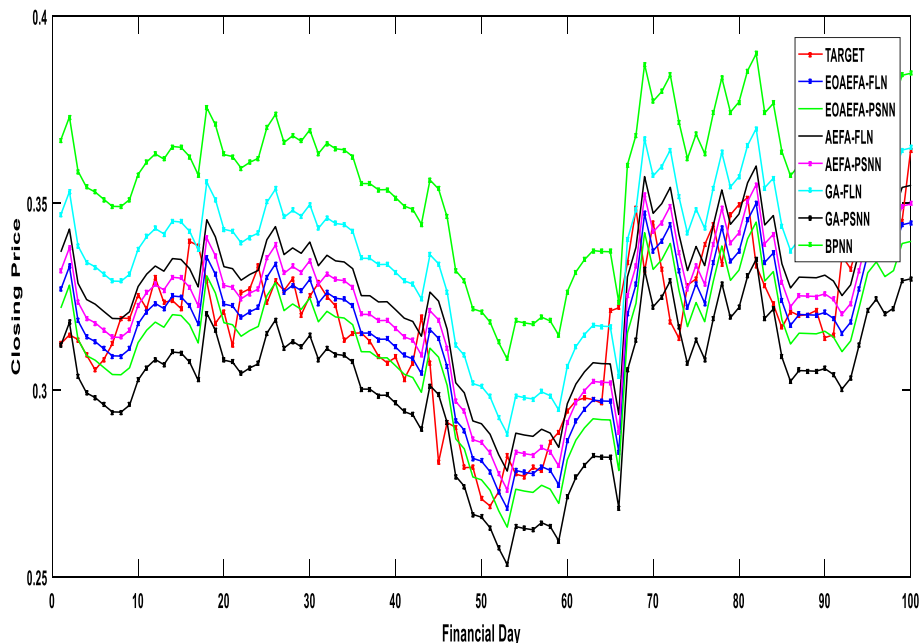


Fig. 13 Prediction plots of seven models from Russell 2000 series

in particular, the EOAEFA-FLN generated the lowest average errors of 0.005833, 0.006013, and 0.006599 from the BTC/USD, EUR/USD, and GBP/USD series, respectively, followed by the EOAEFA-PSNN. However, for the JPY/USD series, the AEFA-PSNN is the best performer with an error of 0.004579. Overall, the EOAEFA-FLN obtained the lowest minimum, mean, median, maximum, standard deviation, and

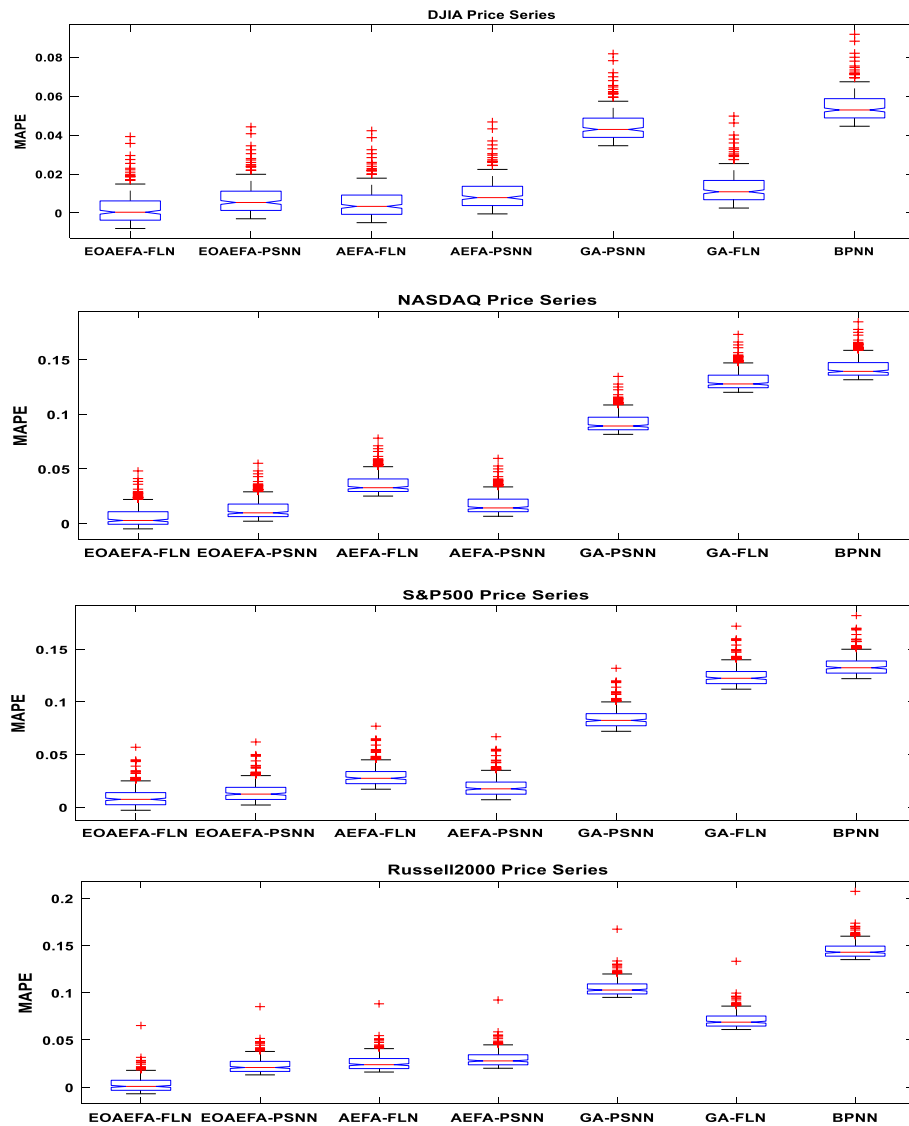


Fig. 14 Box-Whisker plots of MAPE values with seven forecasts and four closing price FTS

interquartile range values, which is evidence of its superior performance in capturing the underlying nonlinearity coupled with the exchange rate FTS. Among the forecasts, the BPNN obtained poor prediction accuracies on the four FTS owing to backpropagation-based learning. The enhanced learning capability of the EOA-EFA was established by these numerical outcomes. Figure 16, 17, 18, 19 depict the predicted versus actual exchange rates from all models for BTC/USD, EUR/USD, GBP/USD, and JPY/USD, respectively. Evidently, the EOA-EFA-FLN and EOA-EFA-PSNN can follow the original FTS patterns more accurately than the others. Although the GA- and basic AEFA-based forecasting seems to follow the original pattern, their prediction values are much larger than those of the proposed models. Similar conclusions can be drawn from the box-whisker plots shown in Fig. 20.

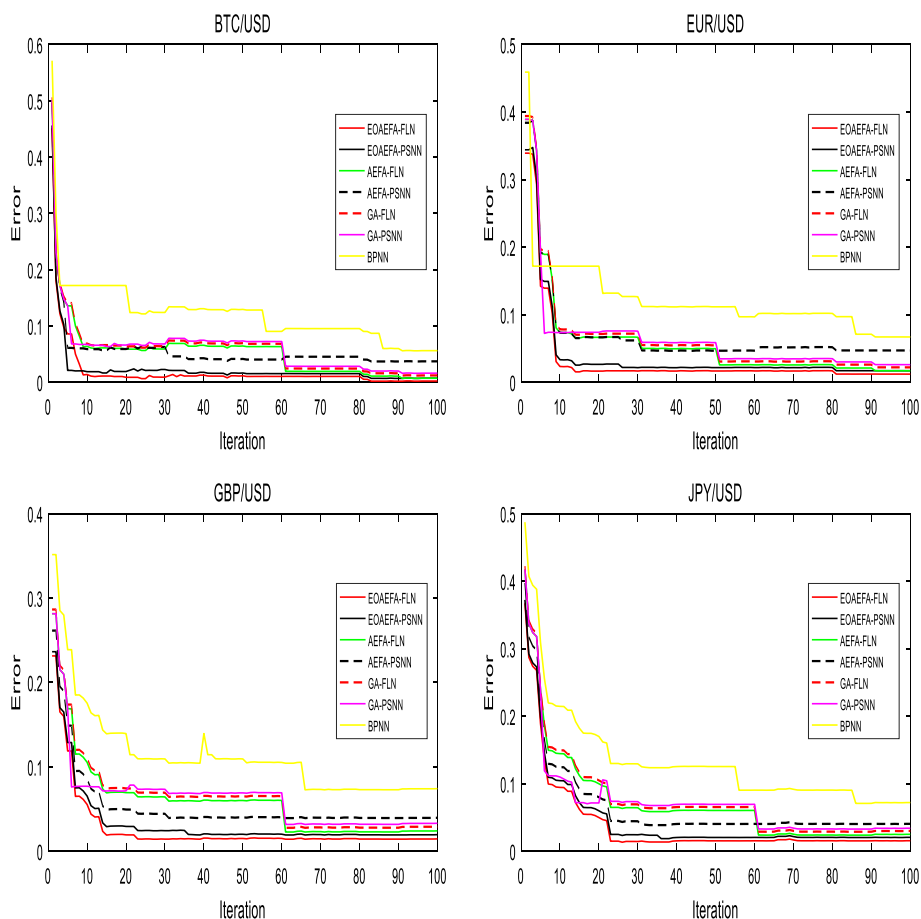


Fig. 15 MAPE convergence rate of seven forecast from four exchange rate FTS

5.5 Analysis of results from the energy prices series.

The proficiency of the proposed forecasts was then exploited to forecast three energy price FTS. The input selection, preprocessing, and model training were conducted in a similar fashion to the previous FTS forecasting. Figure 21 depicts the error convergence rate of all forecasts from the three FTS. Here, the EOA-EFA-FLN converged faster than the others again. Table 8 lists the average error values from 30 independent runs. Values less than 10^{-5} are considered to be zero. For the crude oil and natural gas price series, the EOA-EFA-FLN first again, followed by the EOA-EFA-PSNN. It achieved average errors of 0.003688 and 0.004725 for the crude oil and natural gas price series, respectively. For the weekly coal price series, the EOA-EFA-FLN and EOA-EFA-PSNN obtained the same average error, followed by the AEFA-PSNN-based forecasts. The AEFA- and GA-based predictions were moderate, whereas those of the BPNN were inferior. The prices predicted by the proposed approach were closer to the actual prices, as shown in Figs. 22, 23, 24 for the crude oil, natural gas, and coal price series. Forecasted prices appear to deviate slightly from actual prices in the case of the weekly coal price data. Premature training resulting from insufficient training data (only 50 data points were available) may be the reason for this. However, the direction of movement of all FTS was well retained by the proposed forecasts. The box-whisker plot in Fig. 25 further supports the superiority of the proposed forecasts.

Table 7 MAPE statistics from four exchange rate series

FTS	Forecast	Minimum	Average	Maximum	Median	Standard deviation	Inter-quartile range
BTC/USD	EOAEFA-FLN	0.000000	0.005833	0.033781	0.004161	0.005761	0.006172
	EOAEFA-PSNN	0.000000	0.006705	0.032934	0.005256	0.005814	0.006739
	AEFA-FLN	0.000000	0.007222	0.033934	0.005872	0.005873	0.006689
	AEFA-PSNN	0.000360	0.010731	0.038934	0.009965	0.006399	0.007718
	GA-FLN	0.000000	0.011320	0.043781	0.010485	0.007090	0.007860
	GA-PSNN	0.006219	0.039443	0.068934	0.039680	0.008185	0.008172
	BPNN	0.071219	0.104443	0.133934	0.104680	0.008185	0.008172
EUR/USD	EOAEFA-FLN	0.000000	0.006013	0.031235	0.004345	0.005498	0.006500
	EOAEFA-PSNN	0.000000	0.009508	0.038235	0.008532	0.006559	0.007828
	AEFA-FLN	0.000000	0.006870	0.028212	0.005484	0.005598	0.006666
	AEFA-PSNN	0.000000	0.006214	0.032235	0.004505	0.005599	0.006021
	GA-FLN	0.000830	0.021858	0.046212	0.021617	0.007654	0.009112
	GA-PSNN	0.000385	0.027732	0.052212	0.027617	0.007994	0.009112
	BPNN	0.062765	0.092715	0.117212	0.092617	0.008054	0.009112
GBP/USD	EOAEFA-FLN	0.000000	0.006599	0.032546	0.005009	0.005915	0.006984
	EOAEFA-PSNN	0.000000	0.007158	0.028546	0.005805	0.005752	0.007659
	AEFA-FLN	0.000000	0.007594	0.027546	0.006106	0.005753	0.008092
	AEFA-PSNN	0.000000	0.010590	0.032435	0.009866	0.006396	0.008724
	GA-FLN	0.000000	0.011828	0.042546	0.011144	0.008015	0.009834
	GA-PSNN	0.007454	0.038725	0.062435	0.039009	0.008779	0.010261
	BPNN	0.072454	0.103725	0.127435	0.104009	0.008779	0.010261
JPY/USD	EOAEFA-FLN	0.000000	0.004710	0.019530	0.003627	0.004230	0.004846
	EOAEFA-PSNN	0.000000	0.006805	0.026204	0.006250	0.004590	0.006059
	AEFA-FLN	0.000000	0.007428	0.024530	0.006704	0.005044	0.006124
	AEFA-PSNN	0.000000	0.004579	0.020204	0.003501	0.004125	0.004532
	GA-FLN	0.003796	0.024551	0.042530	0.024368	0.006145	0.006653
	GA-PSNN	0.009796	0.030551	0.048530	0.030368	0.006145	0.006653
	BPNN	0.074796	0.095551	0.113530	0.095368	0.006145	0.006653

Statistical testing and further analysis

To further confirm the benefits of the EOAEFA + HONN-based forecasting, we conducted statistical tests, such as the Wilcoxon signed-rank and Deibold Mariano (DM) tests. In addition, runtime, relative worth, and MAPE reduction percentage analyses were conducted when adopting the proposed model. This section summarizes the outcomes of these tests and analyses.

The forecast performances were compared in terms of computation time. The experiments in this study were conducted using a system with an Intel(R) Core (TM) i7-10750H CPU @ 2.60 GHz, 2.59 GHz, and 16.0 GB of memory in a MATLAB-2016 programming environment. Table 9 summarizes the execution times of all the models s. For input size N and functional expansion unit size FE , an FLN-based model must adjust the $(N \times FE)$ number of weights and one *bias* value. Similarly, for hidden layer width NH , a PSNN-based model must adjust $(N \times NH)$ weights and NH biases. With M hidden neurons and one output neuron, the BPNN model must fine-tune $M \times (N + 1)$ weights and $(M + 1)$ biases. Evidently, the BPNN required the highest running time because of a greater number of adjustable parameters and backpropagation learning.

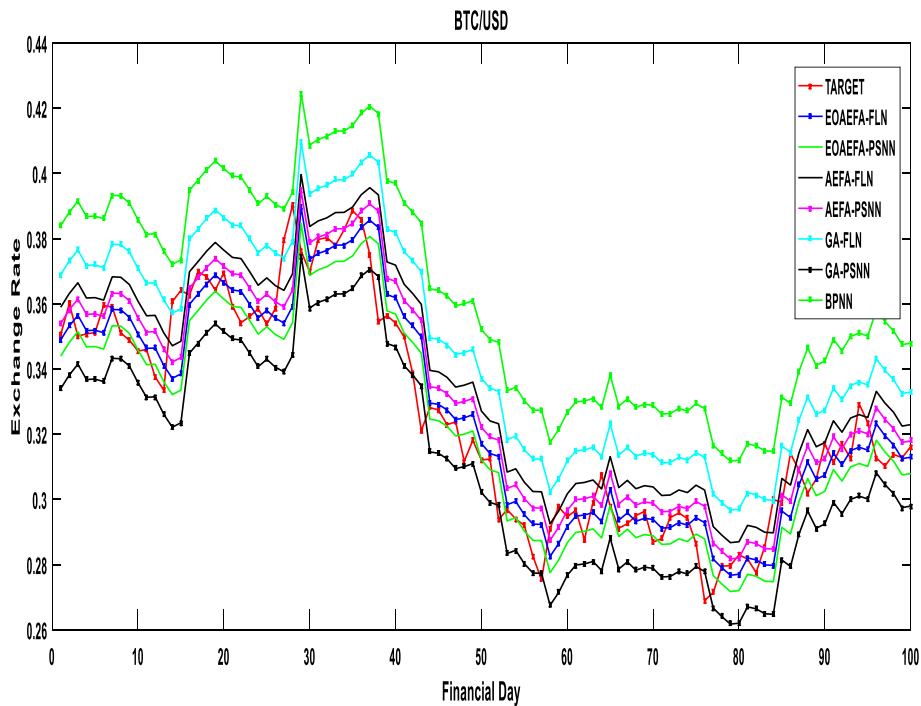


Fig. 16 Prediction plots of seven models from BTC/USD FTS

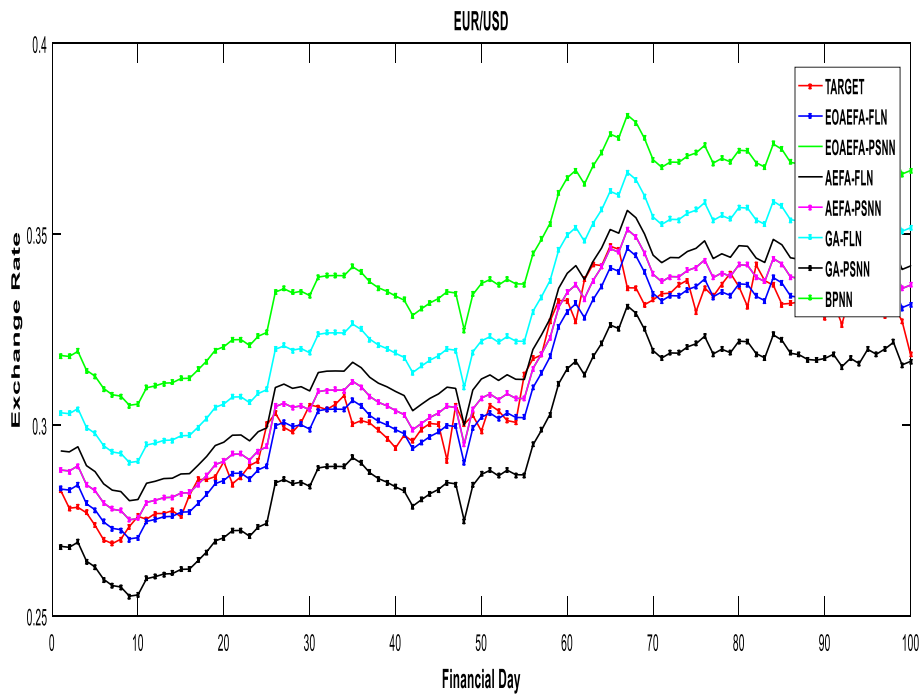


Fig. 17 Prediction plots of seven models from EUR/USD FTS

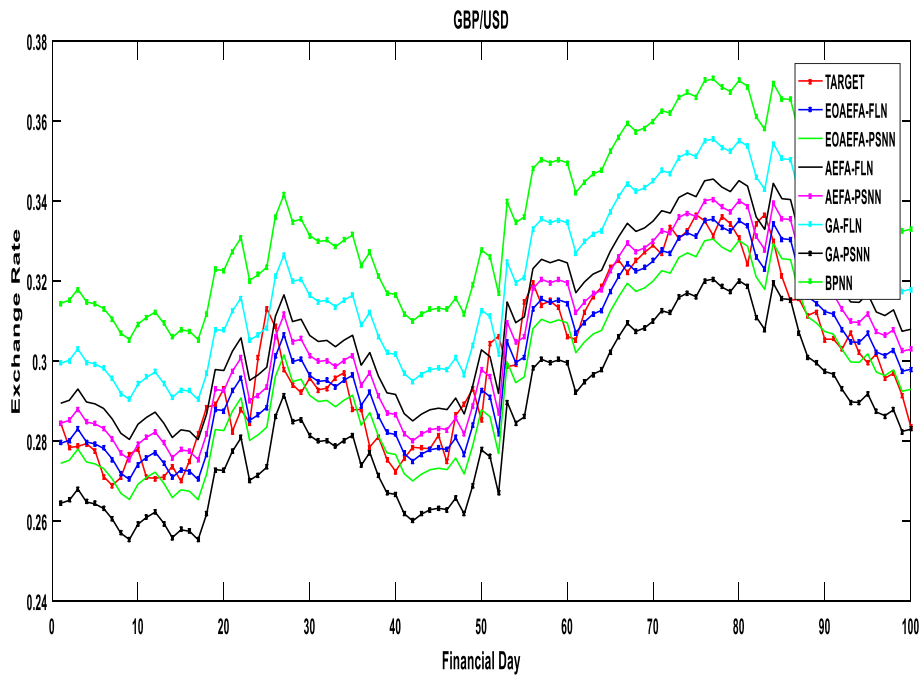


Fig. 18 Prediction plots of seven models from GBP/USD FTS

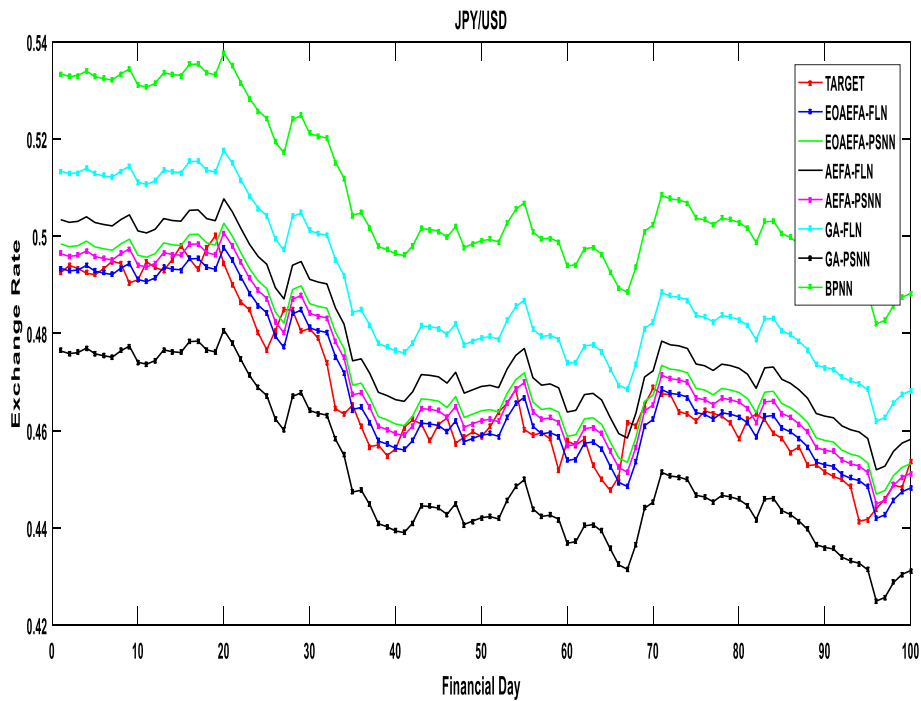


Fig. 19 Prediction plots of seven models from JPY/USD FTS

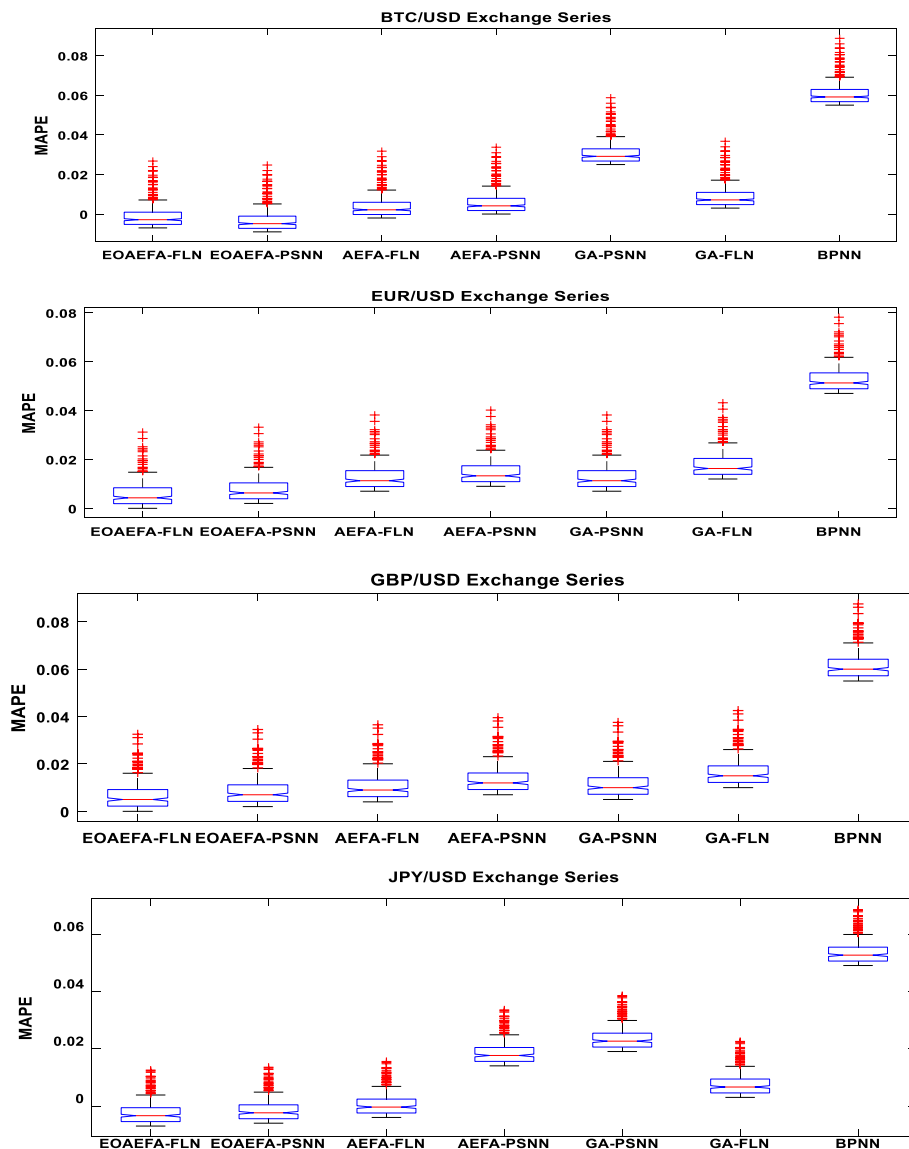


Fig. 20 Box-Whisker plots of MAPE values with seven forecasts and four exchange rate FTS

The run times of the EOA-EFA-FLN and EOA-EFA-PSNN were nearer and slightly greater than those of the basic AEFA-based models because of the inclusion of OBL. However, this can be tolerated as compensation for a higher forecasting accuracy.

The Wilcoxon signed-rank test, which is a paired two-sided test, was conducted as a significance check. The null hypothesis indicated that the variance in the proposed and comparative models originated from a distribution with zero medians. Rejection is indicated by the logical value $h = 1$. The DM test was a twosome comparison of forecasts. This test is used to determine whether the forecasts under consideration are equally acceptable. The null hypothesis states that the two forecasts have an equivalent accuracy, and the alternate hypothesis states that they have different levels of accuracy. If the computed DM statistics lie beyond the critical values, i.e., $-1.965 < DM < 1.965$, the null hypothesis of no variance is rejected. Table 10

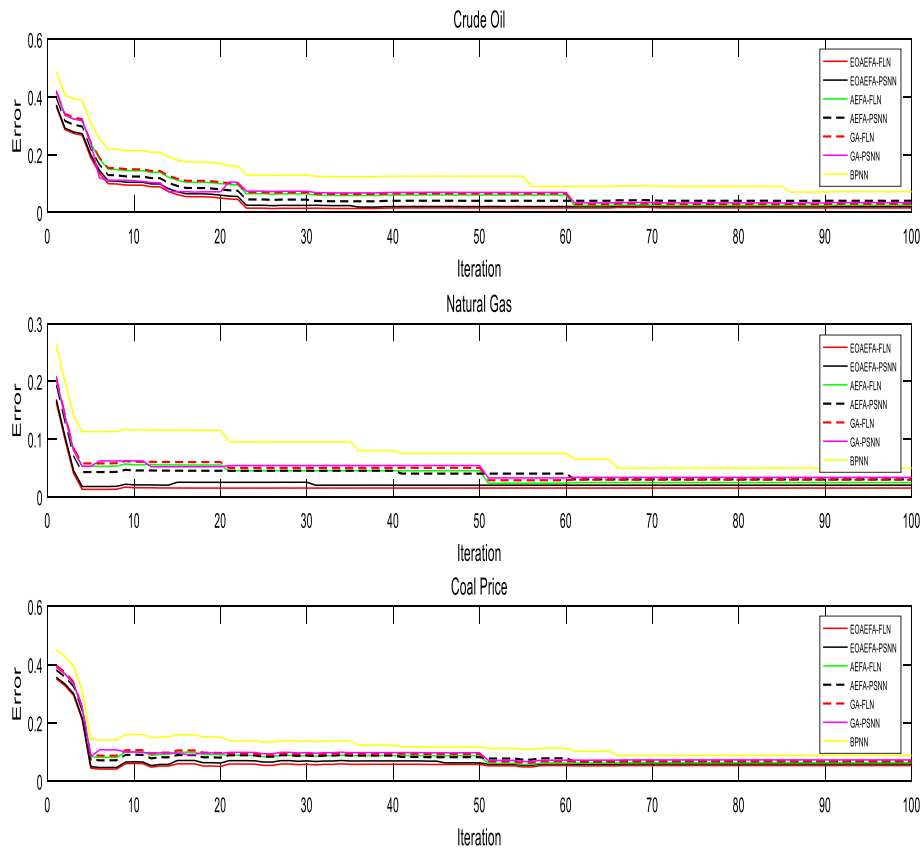


Fig. 21 MAPE convergence rate of seven forecast from three energy price FTS

presents the Wilcoxon signed-rank test (with a 5% significance level) and DM test results from the closing price series. These statistics indicate that the predictions of the proposed forecast significantly differ from those of the other forecasts.

The efficiency of the proposed forecasting approach was observed in the above discussion. In most cases, the proposed forecast achieved the lowest error statistics compared with the other forecasts. To determine the precision of the comparative performance of the EOA-EFA-FLN, another measure, called the relative worth (RW), of the model was considered. This is the average reduction ratio of the prediction error of a particular model using the proposed method over all the FTS. We used the MAPE values to obtain the RWs. The relative worth RW_j of a model over the worst-performing model is defined in Eq. (25). For our calculations, we considered BPNN to be the worst-performing model;

$$RW_j = \frac{1}{N} \sum_{i=1}^N \left(\frac{MAPE_i - MAPE_{ij}}{MAPE_i} \right) \times 100\%, \forall j = 1, 2, 3, \dots, n, \tag{25}$$

where $MAPE_{ij}$ is the forecasting error of the j^{th} model on the i^{th} FTS, $MAPE_i$ is the error of the worst-performing forecast for the same FTS, and N is the number of datasets. Table 11

Table 8 MAPE statistics from three energy price series

FTS	Forecast	Minimum	Average	Maximum	Median	Standard deviation	Inter-quartile range
Crude oil prices	EOAEFA-FLN	0.000000	0.003688	0.044976	0.002342	0.004777	0.003355
	EOAEFA-PSNN	0.000000	0.003853	0.046976	0.002407	0.004996	0.003720
	AEFA-FLN	0.000000	0.006461	0.047155	0.005936	0.004607	0.003972
	AEFA-PSNN	0.000101	0.005710	0.046155	0.005138	0.004589	0.003730
	GA-FLN	0.000000	0.010191	0.054976	0.00934	0.005428	0.004427
	GA-PSNN	0.000871	0.044159	0.086155	0.044712	0.006013	0.004418
	BPNN	0.039024	0.084156	0.126155	0.084712	0.006035	0.004418
Natural gas prices	EOAEFA-FLN	0.000000	0.004725	0.065124	0.002572	0.006203	0.004331
	EOAEFA-PSNN	0.000000	0.005063	0.063124	0.003224	0.006111	0.004232
	AEFA-FLN	0.000000	0.007178	0.070124	0.005692	0.006219	0.005276
	AEFA-PSNN	0.000000	0.006433	0.069124	0.004875	0.006240	0.005346
	GA-FLN	0.003545	0.048444	0.113124	0.048201	0.007784	0.005545
	GA-PSNN	0.000455	0.044446	0.109124	0.044201	0.007776	0.005545
	BPNN	0.074545	0.119444	0.184124	0.119201	0.007784	0.005545
Coal prices	EOAEFA-FLN	0.001084	0.039972	0.197986	0.029520	0.041234	0.035597
	EOAEFA-PSNN	0.000151	0.039996	0.199986	0.030145	0.041718	0.034200
	AEFA-FLN	0.000921	0.040375	0.192986	0.028972	0.039968	0.038383
	AEFA-PSNN	0.000000	0.040254	0.193986	0.028706	0.040216	0.037691
	GA-FLN	0.000000	0.058543	0.149986	0.055251	0.035653	0.054641
	GA-PSNN	0.004082	0.056043	0.153986	0.052871	0.035553	0.053740
	BPNN	0.006337	0.085733	0.173161	0.085697	0.037852	0.055759

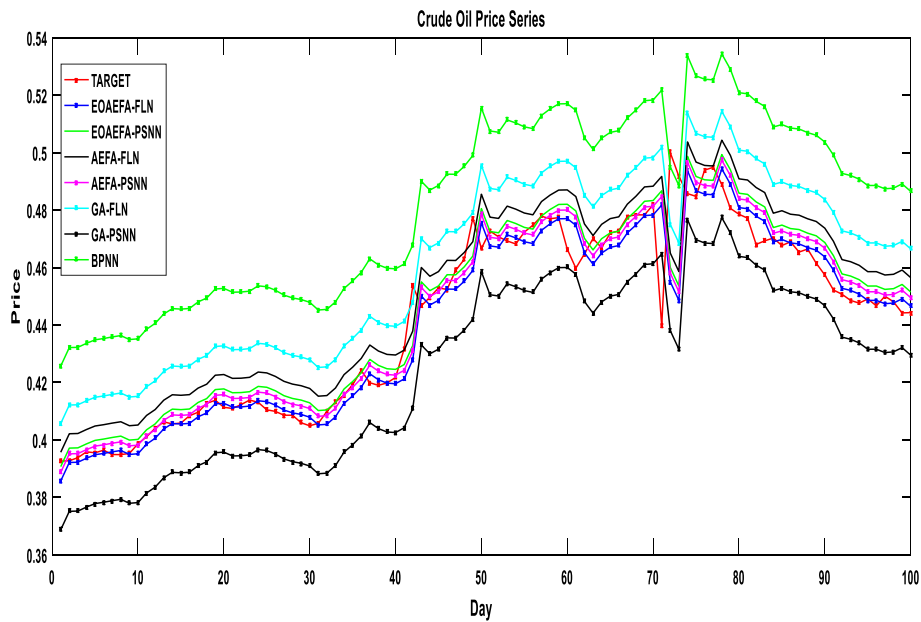


Fig. 22 Prediction plots of seven models from crude oil price series

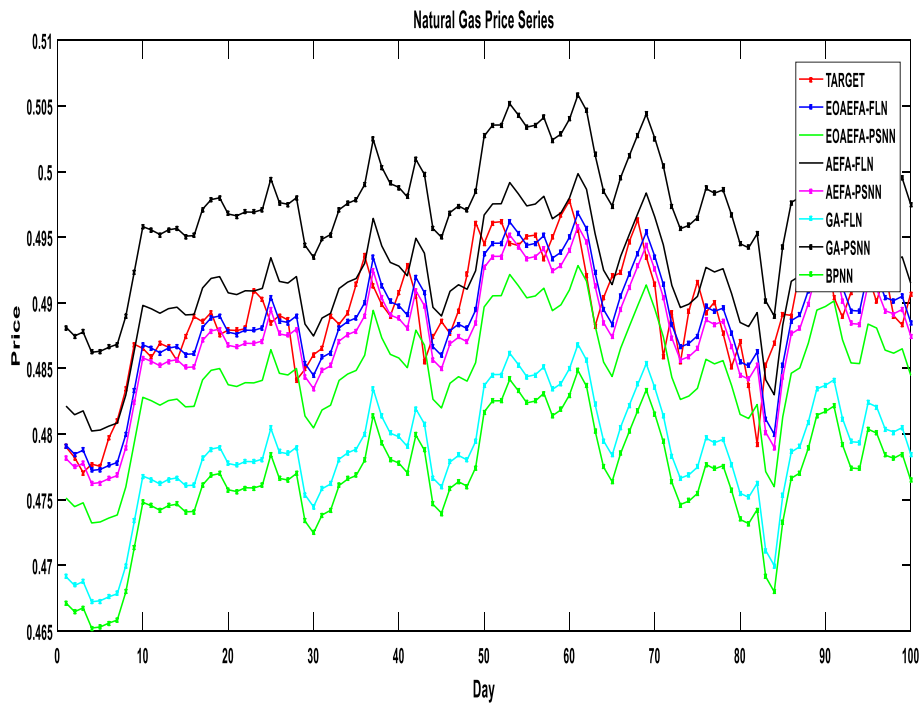


Fig. 23 Prediction plots of seven models from natural gas price series

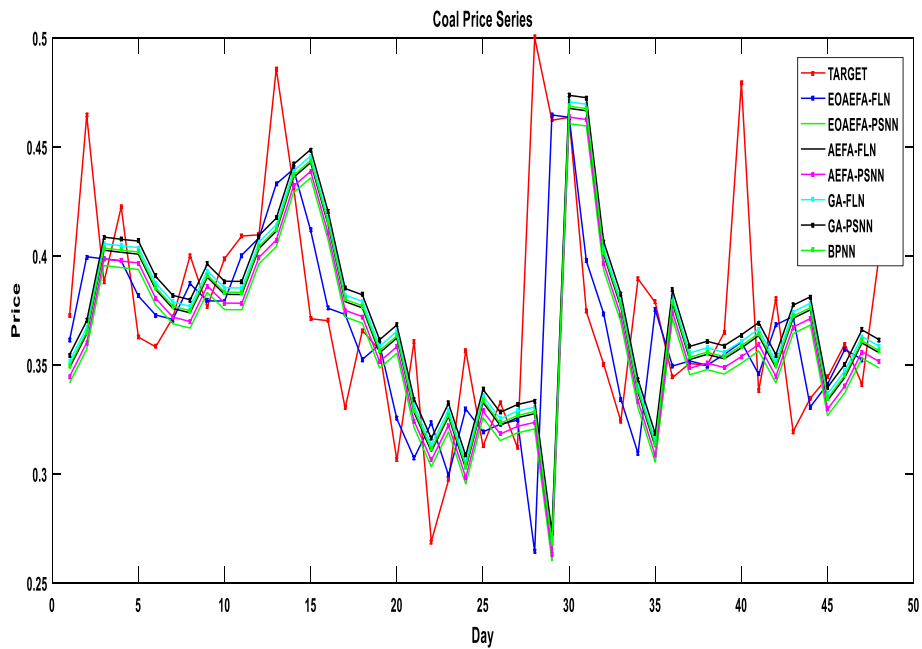


Fig. 24 Prediction plots of seven models from weekly coal price series

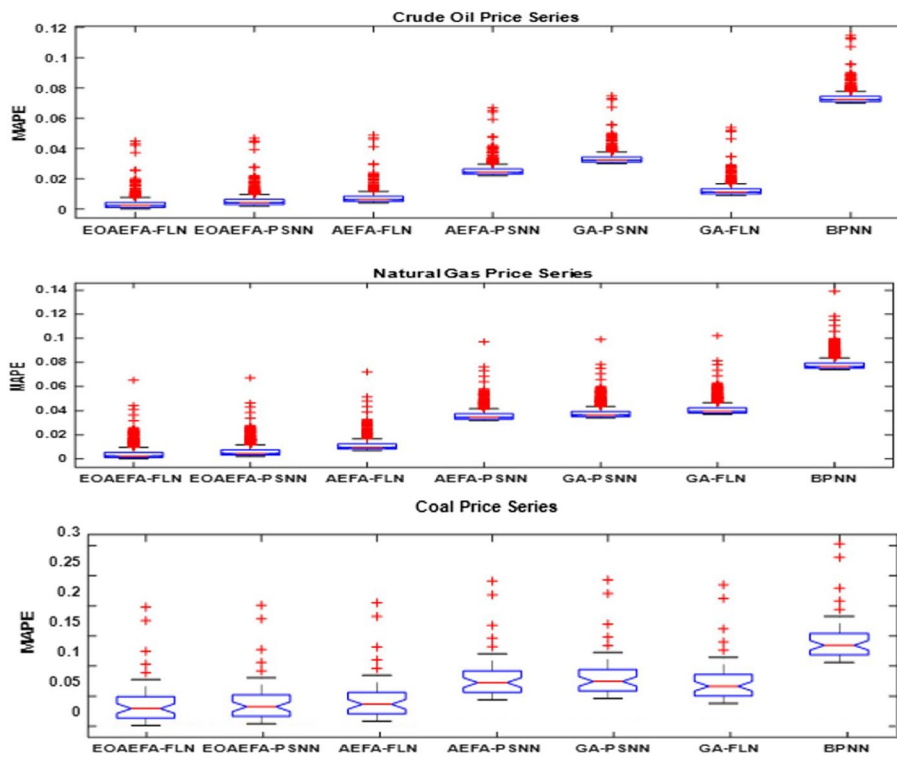


Fig. 25 Box-Whisker plots of MAPE from seven forecast and three FTS

Table 9 Run-time (s) from seven forecasts and eleven FTS

FTS	EOAEFA-FLN	EOAEFA-PSNN	AEFA-FLN	AEFA-PSNN	GA-FLN	GA-PSNN	BPNN
NASDAQ	107.25	100.55	98.53	93.48	113.27	107.88	167.35
DJIA	105.00	110.02	91.80	92.00	115.24	110.55	177.27
S&P500	99.79	101.00	92.49	92.18	108.50	108.00	171.02
RUSSELL 2000	102.66	101.34	97.80	101.05	109.73	110.00	175.15
BTC/USD	105.55	100.39	90.76	97.15	107.28	105.00	205.22
EUR/USD	107.00	104.72	99.07	96.15	112.88	109.83	189.32
GBP/USD	101.47	99.89	91.75	94.55	111.26	108.55	176.72
JPY/USD	105.74	102.77	97.26	98.20	104.73	104.89	170.47
Crude oil prices	122.45	127.00	110.84	112.93	137.48	133.27	228.33
Natural gas prices	125.05	125.66	113.85	115.02	130.65	131.55	220.85
Coal prices	51.96	53.04	51.33	52.07	62.35	60.96	81.55

lists the computed RW values. Based on these statistics, the EOA-EFA-based HONN forecasting has high RW values compared with those of the others.

The model predictions were compared in terms of the percentage reduction in the MAPE (MR) values when adopting the proposed prediction model. This is computed as in Eq. (26). Figure 26 shows the computed MR values from all FTS. Figure 26 presents a good reduction in the MAPE values upon adopting the EOA-EFA-FLN over other forecasts.

$$MR = \frac{(MAPE_{of\ existing\ forecast} - MAPE_{of\ proposed\ forecast})}{MAPE_{of\ existing\ model}} \times 100\% \tag{26}$$

Table 10 Significance test results of the proposed forecast

Methods and FTS	EOAEFA-PSNN	AEFA-FLN	AEFA-PSNN	GA-FLN	GA-PSNN	BPNN
EOAEFA-FLN <i>p</i> and <i>h</i> -values from Wilcoxon signed rank test						
NASDAQ	$p = 1.7522e-1$, $h = 1$	$p = 3.12432e-4$, $h = 1$	$p = 2.0402e-3$, $h = 1$	$p = 3.203e-5$, $h = 1$	$p = 4.216e-4$, $h = 1$	$p = 4.250e-5$, $h = 1$
DJIA	$p = 1.1344e-2$, $h = 1$	$p = 3.0028e-5$, $h = 1$	$p = 2.3074e-3$, $h = 1$	$p = 3.257e-3$, $h = 1$	$p = 3.523e-2$, $h = 1$	$p = 3.134e-3$, $h = 1$
S&P500	$p = 2.3430e-4$, $h = 1$	$p = 0.0035e-3$, $h = 0$	$p = 2.243e-2$, $h = 1$	$p = 2.82017$, $h = 1$	$p = 2.752e-1$, $h = 1$	$p = 2.343e-4$, $h = 1$
RUSSELL 2000	$p = 0.5226e-3$, $h = 1$	$p = 0.003502$, $h = 0$	$p = 3.785e-2$, $h = 1$	$p = 2.266e-2$, $h = 1$	$p = 3.205e-3$, $h = 1$	$p = 3.522e-2$, $h = 1$
<i>p</i> and <i>h</i> -values from DM pairwise test						
NASDAQ	$p = 1.98582$, $h = 1$	$p = 2.14206$, $h = 1$	$p = 2.07284$, $h = 1$	$p = 2.50135$, $h = 1$	$p = -3.00427$, $h = 1$	$p = 3.87091$, $h = 1$
DJIA	$p = 2.60553$, $h = 1$	$p = -2.97426$, $h = 1$	$p = 1.96033$, $h = 0$	$p = 1.9946$, $h = 1$	$p = 2.03294$, $h = 1$	$p = -2.72885$, $h = 1$
S&P500	$p = 1.96273$, $h = 0$	$p = 1.98282$, $h = 1$	$p = 2.10053$, $h = 1$	$p = -2.31042$, $h = 1$	$p = 1.99652$, $h = 1$	$p = 2.83952$, $h = 1$
RUSSELL 2000	$p = -3.30322$, $h = 1$	$p = 2.13216$, $h = 0$	$p = -2.39820$, $h = 1$	$p = 2.40537$, $h = 1$	$p = -2.18215$, $h = 1$	$p = 3.20555$, $h = 1$

Table 11 RW of forecasting models

Model	RW value
EOAEFA-FLN	87.85
EOAEFA-PSNN	83.85
AEFA-FLN	80.02
AEFA-PSNN	78.47
GA-FLN	64.26
GA-PSNN	59.72

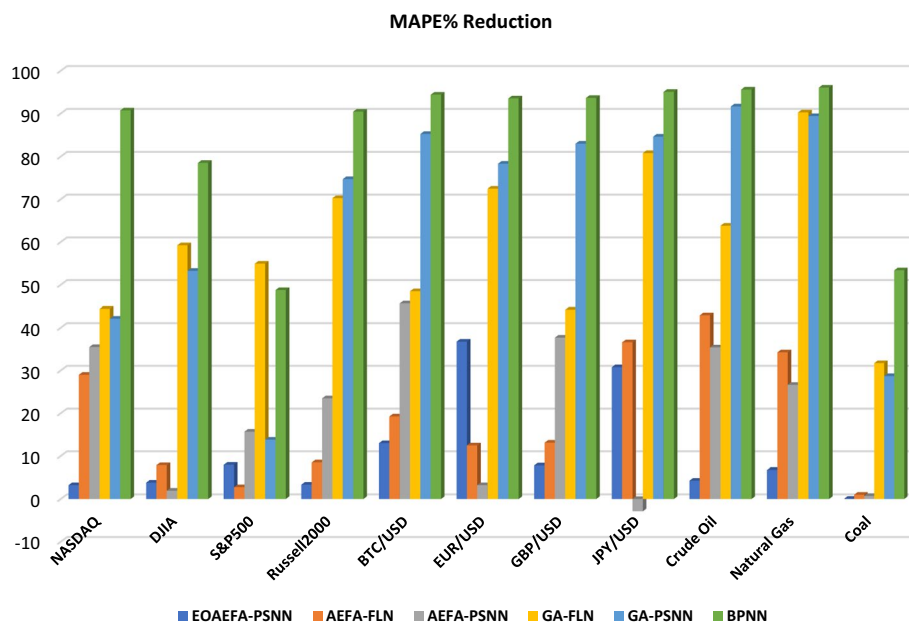


Fig. 26 MAPE reduction on adopting EOAEFA-FLN over other forecasts

Conclusions and future work

The AEFA is a newly developed physics-inspired metaheuristic with a robust development ability, simpler computations, and attainment of global optima with fewer control parameters. Avoiding poor convergences, awhile improving the performance of the basic AEFA and solving real engineering problems remains an open challenge. To address these issues, this study proposes an elitism oppositional learning-based AEFA, called EOAEFA, to intensify the exploration and exploitation abilities of the AEFA. The EOAEFA was used to determine the weights and thresholds of the FLANN and PSNN, forming two hybrid models: EOAEFA-FLN and EOAEFA-PSNN. The proposed methods were evaluated for predicting future patterns of the closing price series of four fast-growing stocks, four exchange rate series from developed economies, and three energy price FTS. The combined effects of the enhanced exploration ability of the EOAEFA, higher fault tolerance capability, and powerful mapping of single-layer trainable weights of the HONNs made the FTS predictions effective and accurate.

To verify the competitiveness of the proposed approach, its predictions were compared with those of AEFA-FLN, AEFA-PSNN, GA-FLN, GA-PSNN, and BPNN-based forecasting. The results show that the EOAEFA + HONNs, particularly the EOAEFA-FLN, obtained accurate predictions for most FTS compared with the others. For the four closing prices FTS, the EOAEFA-FLN generated the lowest MAPE values of 0.010285, 0.010895, 0.012230, and 0.009959. For three of the four exchange rate FTS, it obtained MAPE values of 0.005833, 0.006013, and 0.006599 which are lower than those of the others. Similarly, it generated the lowest MAPE values of 0.003688, 0.004725, and 0.039972 for the three energy prices FTS. The EOAEFA-FLN provided the best forecast, followed by EOAEFA-PSNN. In addition, the EOAEFA-FLN and EOAEFA-PSNN are found 87.85% and 83.85% relative worth compared to the worst performing model respectively. The stronger performance of the EOAEFA-FLN was further established through statistical test results. Further improvements in the EOAEFA performance and its hybridization with other ANNs, such as RNNs and DL methods, are possible extensions of the current study. Moreover, the predictability of the proposed forecast can be further explored in the healthcare and material science engineering domains.

Abbreviations

ANN	Artificial neural network
AEFA	Artificial electric field algorithm
ARIMA	Auto regressive integrated moving average
BPNN	Back propagation neural network
CRO	Chemical reaction optimization
DJIA	Dow Jones Industrial Average
EOAEFA	Elitist opposition based artificial electric field algorithm
EOAEFA-FLN	Elitist opposition based artificial electric field Algorithm FLANN
EOAEFA-PSNN	Elitist opposition based artificial electric field algorithm PSNN
FLANN	Functional link artificial neural network
FTS	Financial time series
FWA	Fireworks algorithm
GA	Genetic algorithm
GWO	Grey wolf optimization
HONN	Higher order neural network
HPUNN	Higher order product unit neural network
LSTM	Long short-term memory
MLP	Multilayer perceptron
MAPE	Mean absolute percentage of errors

NASDAQ	National Association of Securities Dealers Automated Quotations
ObAEF	Opposition based artificial electric field algorithm
OBL	Opposition based learning
OBGA	Opposition based genetic algorithm
OBMA	Opposition based memtic algorithm
PSO	Particle swarm optimization
PSNN	Pi-sigma neural network
PUNN	Product unit neural network
RAFLANN	Rao algorithm based FLANN
RW	Relative worth
SVR	Support vector regression
SCA	Sine–Cosine Algorithm
S&P500	Standard and Poor 500

Acknowledgements

This work was supported by the Yonsei Fellow Program funded by Lee Youn Jae, Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korean government, Ministry of Science and ICT (MSIT) (No. 2020-0-01361, Artificial Intelligence Graduate School Program (Yonsei University); No. 2022-0-00113, Developing a Sustainable Collaborative Multi-modal Lifelong Learning Framework). Prof. Satchidananda Dehuri acknowledge the support of Teachers Associateship for Research Excellence (TARE) Fellowship (No. TAR/2021/00006) of the Science and Engineering Research Board (SERB), Government of India.

Author contributions

SCN: Problem formulation, data collection, implementation, initial draft preparation. SND: Conceptualization, writing and supervision. SBC: Conceptualization, review and supervision. All author(s) read and approved the final manuscript.

Funding

Not applicable.

Availability of data and materials

The datasets analyzed and experimented during the current study are openly available at <https://finance.yahoo.com> and <http://www.eia.doe.gov/>. The source of datasets is highlighted in the article.

Declarations

Competing interests

The author declare that they have no competing interests.

Received: 23 September 2022 Accepted: 25 July 2023

Published online: 02 January 2024

References

- Naik B, Nayak J, Dash PB (2021) Higher order ANN parameter optimization using hybrid opposition-elitism based metaheuristic, *Evol Intell*
- Abd Elaziz M, Oliva D (2018) Parameter estimation of solar cells diode models by an improved opposition-based whale optimization algorithm. *Energy Convers Manage* 171:1843–1859
- Abd Elaziz M, Oliva D, Xiong S (2017) An improved opposition-based sine cosine algorithm for global optimization. *Expert Syst Appl* 90:484–500
- Adegboye OR, Deniz Ülker E (2023) Hybrid artificial electric field employing cuckoo search algorithm with refraction learning for engineering optimization problems. *Sci Rep* 13(1):4098
- Adhikari R, Agrawal RK (2014) A combination of artificial neural network and random walk models for financial time series forecasting. *Neural Comput Appl* 24(6):1441–1449
- Akbulgic O, Bozdoğan H, Balaban ME (2014) A novel hybrid RBF neural networks model as a forecaster. *Stat Comput* 24(3):365–375
- Akman E, Karaman AS, Kuzey C (2020) Visa trial of international trade: evidence from support vector machines and neural networks. *J Manag Anal* 7(2):231–252
- Alanazi A, Alanazi M (2022) Artificial electric field algorithm-pattern search for many-criteria networks reconfiguration considering power quality and energy not supplied. *Energies* 15(14):5269
- Al-Khraisat B, AL-Dmour AS, Al-Maitah K (2021) Artificial electric field algorithm for optimum pmu placement. In: 2021 IEEE green energy and smart systems conference (IGESSC) (pp 1–6). IEEE
- Bai S, Kolter JZ, Koltun V (2018) An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. arXiv preprint [arXiv:1803.01271](https://arxiv.org/abs/1803.01271).
- Bao X, Jia H, Lang C (2019) Dragonfly algorithm with opposition-based learning for multilevel thresholding Color Image Segmentation. *Symmetry* 11(5):716
- Behera AK, Panda M, Nayak SC, Dash C, Kumar S (2022) An artificial electric field algorithm and artificial neural network-based hybrid model for software reliability prediction. In: *Computational intelligence in data mining* (pp 271–279). Springer, Singapore
- Bi J, Zhou Y, Tang Z, Luo Q (2022) Artificial electric field algorithm with inertia and repulsion for spherical minimum spanning tree. *Appl Intell* 52(1):195–214

- Box GE, Jenkins GM, Reinsel GC, Ljung GM (2015) Time series analysis: forecasting and control. Wiley
- Cheng J, Xu P, Xiong Y (2022) An improved artificial electric field algorithm and its application in neural network optimization. *Comput Electr Eng* 101:108111
- Chiroma H, Abdulkareem S, Herawan T (2015) Evolutionary neural network model for West Texas intermediate crude oil price prediction. *Appl Energy* 142:266–273
- Das SR, Mishra D, Rout M (2020) A hybridized ELM-Jaya forecasting model for currency exchange prediction. *J King Saud Univ Comp Inf Sci* 32(3):345–366
- Das S, Nayak SC, Sahoo B (2022) Modelling and forecasting stock closing prices with hybrid functional link artificial neural network. In: *Computational intelligence in data mining* (pp 249–259) Springer, Singapore
- Dash R, Routray A, Dash R, Rautray R (2021a) Designing an efficient predictor model using PSNN and crow search-based optimization technique for gold price prediction. IOS press, Intelligent Decision Technologies
- Dash CSK, Behera AK, Nayak SC, Dehuri S (2021b) QORA-ANN: quasi opposition-based Rao algorithm and artificial neural network for cryptocurrency prediction. In: 2021b 6th International Conference for Convergence in Technology (I2CT) (pp. 1–5). IEEE
- Demirören A, Ekinci S, Hekimoğlu B, Izci D (2021) Opposition-based artificial electric field algorithm and its application to FOPID controller design for unstable magnetic ball suspension system. *Eng Sci Technol Int J* 24(2):469–479
- Demirören A, Hekimoğlu B, Ekinci S, Kaya S (2019) Artificial electric field algorithm for determining controller parameters in AVR system. In: 2019 International Artificial Intelligence and Data Processing Symposium (IDAP) (pp. 1–7). IEEE
- Du H, Wang Z, Zhan WEI, Guo J (2018) Elitism and distance strategy for selection of evolutionary algorithms. *IEEE Access* 6:44531–44541
- Durbin R, Rumelhart DE (1989) Product units: a computationally powerful and biologically plausible extension to back propagation networks. *Neural Comput* 1(1):133–142
- Ecer F, Ardabili S, Band SS, Mosavi A (2020) Training multilayer perceptron with genetic algorithms and particle swarm optimization for modeling stock price index prediction. *Entropy* 22(11):1239
- Ekinci S, Demirören A, Zeynelgil H, Hekimoğlu B (2020) An opposition-based atom search optimization algorithm for automatic voltage regulator system. *J Fac Eng Archit Gazi Univ* 35:1141–1158
- Fama EF (1970) Efficient capital markets: a review of theory and empirical work. *J Financ* 25(2):383–417
- Ghazali R (2005) Higher order neural network for financial time series prediction, Annual Postgraduate Research Conference, School of Computing and Mathematical Sciences, Liverpool John Moores University, UK. <http://www.cms.livjm.ac.uk/research/>
- Giles CL, Maxwell T (1987) Learning, invariance, and generalization in a high-order neural network. *Appl Opt* 26(23):4972–4978
- Guler M, Sahin E (1994) A new higher-order binary-input neural unit: learning and generalizing effectively via using minimal number of monomials. In: *Proceedings of third turkish symposium on artificial intelligence and neural networks* (pp 51–60)
- Guo Y, Han S, Shen C, Li Y, Yin X, Bai Y (2018) An adaptive SVR for high-frequency stock price forecasting. *IEEE Access* 6:11397–11404
- Houssein EH, Hashim FA, Ferahtia S, Rezk H (2021) An efficient modified artificial electric field algorithm for solving optimization problems and parameter estimation of fuel cell. *Int J Energy Res* 45(14):20199–20218
- Hsu MW, Lessmann S, Sung MC, Ma T, Johnson JE (2016) Bridging the divide in financial market forecasting: machine learners vs. financial economists. *Expert Syst Appl* 61:215–234
- Ibrahim RA, Elaziz MA, Oliva D, Cuevas E, Lu S (2019) An opposition-based social spider optimization for feature selection. *Soft Comput* 23(24):13547–13567
- Izci D, Ekinci S, Orenc S, Demirören A (2020) Improved artificial electric field algorithm using nelder-mead simplex method for optimization problems. In: 2020 4th international symposium on multidisciplinary studies and innovative technologies (ISMSIT) (pp 1–5). IEEE
- Jain P, Saxena A (2019) An opposition theory enabled moth flame optimizer for strategic bidding in uniform spot energy market. *Eng Sci Technol Int J* 22(4):1047–1067
- Jerin Leno I, Saravana Sankar S, Ponnambalam SG (2016) An elitist strategy genetic algorithm using simulated annealing algorithm as local search for facility layout design. *Int J Adv Manuf Technol* 84(5):787–799
- Jiang S, Wang Y, Ji Z (2014) Convergence analysis and performance of an improved gravitational search algorithm. *Appl Soft Comput* 24:363–384
- Kar BP, Nayak SK, Nayak SC (2016) Opposition-based ga learning of artificial neural networks for financial time series forecasting. In: *Computational intelligence in data mining, Vol 2* (pp 405–414). Springer, New Delhi
- Kara Y, Boyacioglu MA, Baykan ÖK (2011) Predicting direction of stock price index movement using artificial neural networks and support vector machines: the sample of the Istanbul stock exchange. *Expert Syst Appl* 38(5):5311–5319
- Knowles A, Hussain A, Deredy WE, Lisboa PGJ, Dunis C (2005) Higher-order neural network with Bayesian confidence measure for prediction of EUR/USD exchange rate, Forecasting Financial Markets Conference, 1–3 June, Marseilles, France
- Kou G, Xu Y, Peng Y, Shen F, Chen Y, Chang K, Kou S (2021a) Bankruptcy prediction for SMEs using transactional data and two-stage multiobjective feature selection. *Decis Support Syst* 140:113429
- Kou G, Olgu Akdeniz Ö, Dinçer H, Yüksel S (2021b) Fintech investments in European banks: a hybrid IT2 fuzzy multidimensional decision-making approach. *Financ Innov* 7(1):39
- Leerink L, Giles C, Horne B, Jabri M (1994) Learning with product units. *Advances in neural information processing systems*, 7. Cambridge, MA: MIT Press. pp 537–544
- Li T, Kou G, Peng Y, Philip SY (2021) An integrated cluster detection, optimization, and interpretation approach for financial data. *IEEE Trans Cybern* 52(12):13848–13861
- Mahdavi S, Rahnamayan S, Deb K (2018) Opposition based learning: a literature review. *Swarm Evol Comput* 39:1–23
- Majhi R, Panda G, Sahoo G (2009) Development and performance evaluation of FLANN based model for forecasting of stock markets. *Expert Syst Appl* 36(3):6800–6808

- Nayak SC (2017) Development and performance evaluation of adaptive hybrid higher order neural networks for exchange rate prediction. *Int J Intell Syst Appl* 10(8):71
- Nayak SC (2020) A fireworks algorithm-based Pi-Sigma neural network (FWA-PSNN) for modelling and forecasting chaotic crude oil price time series. *EAI Endorsed Trans Energy Web* 7(28):e2–e2
- Nayak SC (2022) Bitcoin closing price movement prediction with optimal functional link neural networks. *Evol Intel* 15(3):1825–1839
- Nayak SC, Ansari MD (2019b) COA-HONN: cooperative optimization algorithm based higher order neural networks for stock forecasting. *Recent Adv Comput Sci Commun* 2019(12):1–00
- Nayak SC, Misra BB (2019a) A chemical-reaction-optimization-based neuro-fuzzy hybrid network for stock closing price prediction. *Financ Innov* 5(1):1–34
- Nayak SC, Misra BB (2019b) A Chemical reaction optimization based neuro-fuzzy hybrid network for stock closing prices prediction. Springer, *Financial Innovation*
- Nayak SC, Misra BB, Behera HS (2014) Impact of data normalization on stock index forecasting. *Int J Comput Inf Syst Ind Manag Appl* 6(2014):257–269
- Nayak SC, Misra BB, Behera HS (2016) Fluctuation prediction of stock market index by adaptive evolutionary higher order neural networks. *Int J Swarm Intell* 2(2–4):229–253
- Nayak SC, Misra BB, Behera HS (2017a) Artificial chemical reaction optimization of neural networks for efficient prediction of stock market indices. *Ain Shams Eng J* 8(3):371–390
- Nayak SC, Ansari MD (2019a) COA-HONN: cooperative optimization algorithm based higher order neural networks for stock forecasting. *Recent Adv Comput Sci Commun*
- Nayak SC, Misra BB, Behera HS (2015) A pi-sigma higher order neural network for stock index forecasting. In: *Computational intelligence in data mining*. Vol 2 (pp 311–319), Springer, New Delhi
- Nayak SC, Misra BB, Behera HS (2017b) Improving performance of higher order neural network using artificial chemical reaction optimization: a case study on stock market forecasting. In: *Nature-inspired computing: concepts, methodologies, tools, and applications* (pp 1753–1780). IGI Global
- Nayak SC, Misra BB, Behera HS (2018) ACFLN: artificial chemical functional link network for prediction of stock market index. *Evolving Systems*, Springer
- Nayak SC, Dash CSK, Behera AK, Mishra BB (2021) A machine learning approach for estimating compressive strength of concrete structures using an artificial electric field algorithm-based neuro-fuzzy predictor. In: *2021 19th OITS international conference on information technology (OCIT)* (pp 229–233). IEEE
- Opara K, Arabas J (2018) Comparison of mutation strategies in differential evolution—a probabilistic perspective. *Swarm Evolut Comput*
- Pao YH, Takefuji Y (1992) Functional-link net computing: thory, system architecture, and functionalities. *Computer* 25:76–79
- Park SI, Smith MJ, Mersereau RM (2000) Target recognition based on directional filter banks and higher-order neural networks. *Digit Signal Process* 10(4):297–308
- Pradhan M, Roy PK, Pal T (2018) Oppositional based grey wolf optimization algorithm for economic dispatch problem of power system. *Ain Shams Eng J* 9(4):2015–2025
- Dash R, Rautray R, Dash R (2019) Utility of a shuffled differential evolution algorithm in designing of a pi-sigma neural network based predictor model. *Applied Computing and Informatics*, Emerald
- Sahu KK, Sahu SR, Nayak SC, Behera HS (2016) Forecasting foreign exchange rates using CRO based different variants of FLANN and performance analysis. *Int J Comput Syst Eng* 2(4):190–208
- Sahu KK, Nayak SC, Behera HS (2019) Towards designing and performance analysis of evolving higher order neural networks for modeling and forecasting exchange rate time series data. In: *Proceedings of ICETIT 2019*, Springer
- Selem SI, El-Fergany AA, Hasanien HM (2021) Artificial electric field algorithm to extract nine parameters of triple-diode photovoltaic model. *Int J Energy Res* 45(1):590–604
- Shadbolt N (2004) Nature-inspired computing. *IEEE Intell Syst* 19(1):2–3
- Shekhawat S, Saxena A (2020) Development and applications of an intelligent crow search algorithm based on opposition-based learning. *ISA Trans* 99:210–230
- Shin Y, Ghosh J (1991) The pi–sigma network: an efficient higher-order neural network for pattern classification and function approximation. In: *International joint conference on neural networks*
- Shin Y, Ghosh J (1995) Ridge polynomial networks. *IEEE Trans Neural Netw* 6(3):610–622
- Sindhu R, Ngadiran R, Yacob YM, Zahri NAH, Hariharan M (2017) Sine–cosine algorithm for feature selection with elitism strategy and new updating mechanism. *Neural Comput Appl* 28(10):2947–2958
- Subhranginee D, Nayak SC, Sahoo B (2021) Towards crafting optimal functional link artificial neural networks with RAO algorithms for stock closing prices prediction. *Comput Econ*
- Tizhoosh HR (2005) Opposition-based learning: a new scheme for machine intelligence. In *International conference on computational intelligence for modelling, control and automation and international conference on intelligent agents, web technologies and internet commerce (CIMCA-IAWTIC'06)* (Vol. 1, pp. 695–701). IEEE.
- Tubishat M, Idris N, Shuib L, Abushariah MA, Mirjalili S (2020) Improved Salp swarm algorithm based on opposition based learning and novel local search algorithm for feature selection. *Expert Syst Appl* 145:113122
- Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser Ł, Polosukhin I (2017) Attention is all you need. *Adv Neural Inf Process Syst*, 30
- Wang Z, Fang J, Liu X (2008) Global stability of stochastic high-order neural networks with discrete and distributed delays. *Chaos Solutions Fractals* 36(2):388–396
- Yadav A (2019) AEFA: artificial electric field algorithm for global optimization. *Swarm Evol Comput* 48:93–108
- Yadav A, Kumar N (2020) Artificial electric field algorithm for engineering optimization problems. *Expert Syst Appl* 149:113308
- Yadav A, Kumar N (2019) Application of artificial electric field algorithm for economic load dispatch problem. In: *International conference on soft computing and pattern recognition* (pp 71–79). Springer, Cham

- Yildiz BS (2020) Robust design of electric vehicle components using a new hybrid salp swarm algorithm and radial basis function-based approach. *Int J Veh Des* 83(1):38–53
- Yildiz BS, Pholdee N, Panagant N, Bureerat S, Yildiz AR, Sait SM (2022a) A novel chaotic Henry gas solubility optimization algorithm for solving real-world engineering problems. *Eng Comput* 38(2):871–883
- Yildiz BS, Kumar S, Pholdee N, Bureerat S, Sait SM, Yildiz AR (2022b) A new chaotic Lévy flight distribution optimization algorithm for solving constrained engineering problems. *Expert Syst* 39(8):e12992
- Yildiz BS, Pholdee N, Bureerat S, Yildiz AR, Sait SM (2021) Robust design of a robot gripper mechanism using new hybrid grasshopper optimization algorithm. *Expert Syst* 38(3):e12666
- Yildiz BS, Pholdee N, Bureerat S, Yildiz AR, Sait SM (2022) Enhanced grasshopper optimization algorithm using elite opposition-based learning for solving real-world engineering problems. *Eng Comput* 38(5):4207–4219
- Yoon J (2020) Forecasting of real GDP growth using machine learning models: gradient boosting and random forest approach. *Comput Econ*, 1–19
- Zhang GP (2003) Time series forecasting using a hybrid ARIMA and neural network model. *Neurocomputing* 50:159–175
- Zheng H, Gao J, Xiong J, Yao G, Cui H, Zhang L (2022) An enhanced artificial electric field algorithm with sine cosine mechanism for logistics distribution vehicle routing. *Appl Sci* 12(12):6240
- Zhou Y, Hao JK, Duval B (2017) Opposition-based memetic search for the maximum diversity problem. *IEEE Trans Evol Comput* 21(5):731–745

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Submit your manuscript to a SpringerOpen[®] journal and benefit from:

- ▶ Convenient online submission
- ▶ Rigorous peer review
- ▶ Open access: articles freely available online
- ▶ High visibility within the field
- ▶ Retaining the copyright to your article

Submit your next manuscript at ▶ [springeropen.com](https://www.springeropen.com)
