Sustainable Cities and Society 66 (2021) 102669

Contents lists available at ScienceDirect

Sustainable Cities and Society

journal homepage: www.elsevier.com/locate/scs

COVID-19 cases prediction by using hybrid machine learning and beetle antennae search approach

Miodrag Zivkovic [a], Nebojsa Bacanin [a,*], K. Venkatachalam [b], Anand Nayyar [c,d], Aleksandar Djordjevic [a], Ivana Strumberger [a], Fadi Al-Turjman [e]

[a] Singidunum University, Danijelova 32, 11000 Belgrade, Serbia
[b] School of Computer Science and Engineering, VIT Bhopal University, Bhopal, India
[c] Graduate School, Duy Tan University, Da Nang 550000, Viet Nam
[d] Faculty of Information Technology, Duy Tan University, Da Nang, Viet Nam
[e] Research Centre for AI and IoT, Department of Artificial Intelligence Engineering, Near East University, 99138 Nicosia, Mersin 10, Turkey

ARTICLE INFO

Keywords:
COVID-19
Beetle antennae search
ANFIS
Machine learning
Swarm intelligence
Enhanced
Prediction

ABSTRACT

The main objective of this paper is to further improve the current time-series prediction (forecasting) algorithms based on hybrids between machine learning and nature-inspired algorithms. After the recent COVID-19 outbreak, almost all countries were forced to impose strict measures and regulations in order to control the virus spread. Predicting the number of new cases is crucial when evaluating which measures should be implemented. The improved forecasting approach was then used to predict the number of the COVID-19 cases. The proposed prediction model represents a hybridized approach between machine learning, adaptive neuro-fuzzy inference system and enhanced beetle antennae search swarm intelligence metaheuristics. The enhanced beetle antennae search is utilized to determine the parameters of the adaptive neuro-fuzzy inference system and to improve the overall performance of the prediction model. First, an enhanced beetle antennae search algorithm has been implemented that overcomes deficiencies of its original version. The enhanced algorithm was tested and validated against a wider set of benchmark functions and proved that it substantially outperforms original implementation. Afterwards, the proposed hybrid method for COVID-19 cases prediction was then evaluated using the World Health Organization's official data on the COVID-19 outbreak in China. The proposed method has been compared against several existing state-of-the-art approaches that were tested on the same datasets. The proposed CESBAS-ANFIS achieved R^2 score of 0.9763, which is relatively high when compared to the R^2 value of 0.9645, achieved by FPASSA-ANFIS. To further evaluate the robustness of the proposed method, it has also been validated against two different datasets of weekly influenza confirmed cases in China and the USA. Simulation results and the comparative analysis show that the proposed hybrid method managed to outscore other sophisticated approaches that were tested on the same datasets and proved to be a useful tool for time-series prediction.

1. Introduction

The recently discovered coronavirus SARS-CoV-2, known by the name COVID-19 (Coronaviridae Study Group of the International et al., 2020), is a novel respiratory virus, which was initially detected in humans in December 2019, in Wuhan, China (Chan et al., 2020; Yadav & Saxena, 2020). Since then, the virus has spread worldwide, affecting more than 200 countries, with the number of reported cases rising to over 13 million infected people, and the number of deaths rising to 570,000 as the mid of July 2020. Since it is a novel virus, scientists and epidemiologists still have a lot to learn about it. One fact that remains unknown is the number of people with the very mild condition or even asymptomatic infection, and how infectious those people actually are. Estimates are varying from 5 to even 50 people without symptoms for every reported case.

The precise scale of the COVID-19 outbreak is not known at the moment, and it will take months, maybe even years to gather all relevant data to give a precise estimate. This will include massive testing on the population which every country will perform to see how many people are actually infected. However, this information is neither available now, nor it was available at the moment of the outbreak.

* Corresponding author.
E-mail address: nbacanin@singidunum.ac.rs (N. Bacanin).

https://doi.org/10.1016/j.scs.2020.102669
Received 25 July 2020; Received in revised form 25 November 2020; Accepted 14 December 2020
Available online 30 December 2020
2210-6707/© 2020 Elsevier Ltd. All rights reserved.

The first estimations from the World Health Organization (WHO) were that the novel coronavirus is extremely contagious and dangerous (World Health Organization et al., 2020). Within the first three months of the pandemic, the virus had spread over all continents and reached almost every country in the world. Most of the countries officials worldwide declared the state of emergency and enforced regulations regarding social distancing, and many other relevant control measures in order to try and control the virus spread and minimize the number of deaths (Sohrabi et al., 2020; Spinelli & Pellino, 2020). The main goal was to limit the number of infected individuals so that the health system does not get overwhelmed by the people with serious respiratory illness who would require intensive care in hospitals. As a result, airports, schools, faculties, public transport, and many businesses were shut down across the world, and the people were encouraged to practice social distancing and work from home if possible. Nevertheless, some countries were affected by the virus much severely than other countries, which have translated unfortunately in a greater number of deaths.

The officials had widely utilized various epidemiological models (Morens, Taubenberger, Harvey, & Memoli, 2010; Rypdal & Sugihara, 2019; Scarpino & Petri, 2019) to try and estimate the outbreak, identify and estimate the peak of the epidemic as early as possible, and to try predicting the number of potential deaths. Based on these prediction models, the officials decided what measures must be taken in order to control the outbreak, suggested new policies, and also assessed the effectiveness of the measures that were already in place. Therefore, the accuracy of the outbreak prediction model which is being used is critical in order to obtain relevant insight into the possible spread of the virus and death toll of the disease.

As a matter of fact, COVID-19 is not the first coronavirus that has threatened humanity in the past twenty years. The first virus outbreak was the SARS in 2003, followed by the MERS outbreak in 2012. In the past two decades, there were several other disease outbreaks around the world, including Ebola, swine flu, H1N1 flu, the previously mentioned SARS and MERS, and the most recent Zika virus. These outbreaks led to the development of novel and advanced epidemiological models, which were able to predict the outbreaks with high accuracy. Unfortunately, the COVID-19 pandemic has shown a non-linear and very complex nature, as it has been shown in Ivanov (2020).

The novel coronavirus outbreak has also exhibited a lot of differences compared to the other previous outbreaks, which had put in doubt the practical ability of the existing models to deliver accurate predictions and results. The COVID-19 outbreak still has multiple unknown variables which are influencing the spread of the virus — the complex and varying behavior of the population in different countries, different approaches of the governments and officials when applying the measures to contain the virus spread, declared a state of emergency to name the few. These unknown variables had decreased current models performances drastically (Scarpino & Petri, 2019). Some of the more recent models have included the influence of social distancing, quarantine, and curfew into their outbreak prediction, i.e. Zhan et al. (2019) and Rypdal and Sugihara (2019).

The overview of the recent literature which considers the prediction of the virus spread shows a significant amount of research currently going on about this hot topic. Most of the recent research focuses on the estimation of the number of infected people, serious cases (infected individuals who must be taken care of in intensive care units), and fatalities. This kind of research is extremely important for the current outbreak, although the virus has already shown some signs of slowing down in some countries especially in Europe and eastern Asia, where the outbreak control measures have already been relaxed to some extent.

The virus is currently raging in North and South America, with the number of reported infections showing that India and Russia particularly have also been affected a lot, so one can safely say that we are still far away from the global pandemic suppression. This research is also important for the future, as no one is certain whether or not there will be a second wave later this year/or next year, and if there is a second wave, would it be more or less dangerous and lethal than the first wave. Secondly, this research can help in predicting the outbreak of some completely new disease in the following years.

The majority of the recent papers deal with the prediction models. Outbreak prediction with the machine learning approach was discussed in Ardabili et al. (2020). It investigates a wide range of machine learning models and outlines two models that have shown promising results (MLP or multi-layered perceptron, and ANFIS — adaptive network-based fuzzy inference system). The conclusion of this research suggests that machine learning can be used effectively to model the outbreak of the disease. This approach was later exemplified on the case of Hungary (Pinter, Felde, Mosavi, Ghamisi, & Gloaguen, 2020), in order to demonstrate the potential of the machine learning approach and to set a path for future research. The research presented in Suzuki and Suzuki (2020) utilize machine learning approach to estimate the number of reported cases in each province of South Korea, by employing a combination of XGBoost and MultiOutputRegressor as a machine learning model. Alternative machine learning approach was conducted in Liu et al. (2020), by combining disease estimates from mechanistic models with digital traces, in order to reliably forecast COVID-19 activity in the Chinese provinces at near real-time. More precisely, the proposed method was able to produce stable and accurate forecasts 2 days ahead of the current time. This was done by combining inputs from official health reports from Chinese Center Disease for Control and Prevention, COVID-19-related internet search activity, news, and media activity, and daily forecasts of COVID-19 activity from GLEAM, an agent-based mechanistic model.

In the modern world, a large number of cities developed into smart cities, with significant application of the Internet of Things (IoT) (Silva, Khan, & Han, 2018). The rapid growth of the population in the cities, together with the urbanization process required introducing new ways to handle this process with minimal consequences to the environment and lifestyle commodity of the citizens. Smart cities evolved through intensive application of IoT, and today the city operations are supported in an intelligent manner, while minimizing the human interaction. The concept of the smart and sustainable city is founded on a basis of a large number of interdisciplinary sciences, working together to achieve ecologically and technologically advanced environment (Bibri & Krogstie, 2017). These cities are almost exclusively found in technologically advanced and rich countries. However, even in the most developed countries, it was obvious that no one was prepared for a global pandemic of COVID-19 proportions. Lock-downs and curfews were necessary in the smart cities. Additionally, schools, restaurants and small business were closed for months. As a result, various research are focused on the social distancing rules and required sufficient ventilation in the buildings, in order to try to bring the lifestyle of the citizens as close to normal as possible while preventing the virus transmission (Sun & Zhai, 2020). The lessons learned during the COVID-19 will have a huge impact on the architecture and urbanism, which will never be the same after the pandemic due to the fear of infection (Megahed & Ghoneim, 2020). Additionally, smart cities will be including new technology for different threats like COVID-19 in the future. The research presented within this paper aimed to show the sophisticated mechanism for predicting the number of new cases of infection on a city level. The simulations conducted in this paper were based on the available dataset from China and other datasets were used for the purpose of the comparative analysis. The proposed method can be easily implemented in smart cities to address any future pandemic situation and help predicting the number of the confirmed cases, which will help in deciding which measures need to be taken and at what time to save lives.

The basic research question, as well as motivation behind the research proposed in this paper, can be defined as follows: *Is it possible to*

further improve current time-series prediction (forecasting) algorithms based on hybrids between machine learning and nature-inspired algorithms?.

To accomplish this goal, an enhanced version of the recently developed beetle antennae search (BAS) algorithm has been adopted for updating parameters of the adaptive neuro-fuzzy inference system (ANFIS) machine learning method. Antecedent and conclusion ANFIS parameters have been taken into the consideration, while the type of membership function was not subject to the optimization process.

Moreover, during practical simulations with the original BAS algorithm, that belongs to the family of swarm intelligence, some deficiencies can be observed. Therefore, for the purpose of this research, first, the basic BAS algorithm has been improved and then both algorithms were tested on the standard set of unconstrained benchmarks to validate enhancements. Next, a framework based on the ANFIS trained by the improved BAS algorithm has been employed to create a prediction model for the virus outbreak anticipation. The main goal of this research is to enhance the prediction accuracy of new cases of COVID-19. The secondary objective is to try to improve the original BAS algorithm by addressing its deficiencies.

Due to the fact that COVID-19 is the most important and urgent global challenge that human beings face, the proposed method has been tested on the COVID-19 dataset from China. Moreover, to show that the proposed method can be successfully applied in predicting time-series of any other disease, additional experiments have been performed with the dataset of weekly confirmed cases of influenza. One additional experiment has been performed to predict if climate and environment variables, such as population density, can have an effect on the infection rate.

The rest of the paper is organized as follows: Section 2 gives an overview of the ANFIS and swarm intelligence metaheuristics application in solving various NP-hard problems, Section 3 provides insights of ANFIS method. In Section 4, details of the original and improved BAS algorithm have been given, as well as of the proposed ANFIS framework implementation. Section 5 presents the simulation results and discussion, while Section 6 provides a conclusion and final remarks of this research along with the recommendations for the future work.

2. Background and related work

This section presents a literature overview of the ANFIS model and swarm intelligence applications in solving various real-life problems. This section also discusses how swarm intelligence can be applied to the optimization of the ANFIS model parameters.

The ANFIS belongs to the group of artificial intelligence techniques, and it merges artificial neural networks with fuzzy inference systems. Its structure allows it to be used in the process of modeling a large number of systems from various application domains. Neuro-fuzzy systems have been applied to solving many real-world problems. ANFIS was introduced by Jang in 1993 (Jang, 1993), and it is considered to be one of the most popular neuro-fuzzy systems. It combines the characteristics and advantages of both artificial neural networks and fuzzy inference systems, therefore providing a firm background for problem identification and modeling. As it can be seen from the available literature, ANFIS has been widely used in time series forecasting. Some of the fields where ANFIS was successfully applied include traffic control, medical systems, economic data, image processing, feature extraction, forecasting, etc Karaboga and Kaya (2019a).

By reviewing the most recent ANFIS publications it can be seen that many successful ANFIS implementations exist. For example, in Naderloo et al. (2012), ANFIS was applied to predict crop yield, based on the different energy inputs. The ANFIS was also used to estimate the relative viscosity of nanofluids (Baghban, Jalali, Shafiee, Ahmadi, & Chau, 2019). In Harandizadeh, Toufigh, and Toufigh (2019), ANFIS approach was used to estimate the bearing capacity of piles, with promising results. Another recent publication shows an approach that employs a hybrid ANFIS model for forest fire probability prediction (Jaafari, Zenner, Panahi, & Shahabi, 2019).

The ANFIS was also used numerous times for disease diagnosis and spread forecasting. The application of neuro-fuzzy systems in forecasting Measles cases in Ethiopia was discussed in Uyar, Ilhan, Iseri, and Ilhan (2019). Other applications of ANFIS for disease forecasts include: Hepatitis C virus epidemic (Khodaei-mehr, Tangestanizadeh, Vatankhah, & Sharifi, 2018), tuberculosis (Mohammed, Ahmed, Al-Mousawi, Azeez, et al., 2018; Uçar, Karahoca, & Karahoca, 2013), and finally, COVID-19. The COVID-19 related applications include forecasting the confirmed cases of the COVID-19 in China (Al-qaness, Ewees, Fan, & Aziz, 2020), outbreak prediction (Ardabili et al., 2020) and prediction case-study on the state of Hungary (Pinter et al., 2020). Since the COVID-19 is a relatively new challenge, only a few studies have been found from this domain.

One of the greatest issues and challenges in machine learning algorithms is to establish optimal or near-optimal values of its parameters for tackling a specific problem. Unfortunately, there is no universal rule and to solve each specific problem, a different set of parameters values should be determined. Establishing optimal or near-optimal values of these parameters is an NP-hard task and for its solving metaheuristics approaches could be applied.

NP-hard problems cannot be solved within the polynomial time if only traditional (deterministic) methods are used. NP-hard problems have considerable practical importance and belong to the domain of the theory of computational complexity, which plays a central role in modern computer science. Practical NP-hard problems can be found in machine learning, cloud computing, wireless sensor networks, software, and hardware design and operations research, to name the few. To solve this kind of problems in a reasonable amount of time, the stochastic approach can be applied.

Metaheuristics are the form of the stochastic approaches, and their goal is to find an approximate solution which is good enough (not necessarily the best solution), within the reasonable time (Strumberger, Bacanin, & Tuba, 2017; Tuba, Bacanin, & Alihodzic, 2015; Tuba, Strumberger, Bacanin, & Tuba, 2018). Recently, metaheuristic algorithms have been utilized in solving a large number of NP-hard problems (Strumberger, Minovic, Tuba, & Bacanin, 2019). One of the most prominent families of metaheuristics is bio-inspired (nature-inspired) algorithms. In general, bio-inspired metaheuristics can be divided into two large distinctive groups. The first group of algorithms is known under the name evolutionary algorithms (EA), while swarm intelligence algorithms represent the second group.

The EA mimics the process of natural selection, which can be defined as the survival of the most fit. The most fit individuals are selected for breeding to produce offspring for the next generation. Therefore, natural selection starts by selecting the most fit individuals from a given population. These individuals breed and produce the offspring which are added to the next generation and which will inherit the beneficial characteristics from the parents. The offspring is assumed to be better than its parents, resulting in a better chance for survival. As the process iterates, in the end, it will result in a generation of the most fit individuals. This logic can be directly applied to a search problem. A set of solutions for a given problem can be observed and the best solutions can be selected.

The most important example of the evolutionary algorithms is a genetic algorithm (GA) (Goldberg, 1989). The GA was used to solve numerous NP-hard real-life problems in the past, including scheduling and load balancing in the cloud computing (Wang, Liu, Chen, Xu, & Dai, 2014; Zhan, Zhang, Ying-Lin, Gong, & Zhang, 2014), designing convolutional neural networks (Baldominos, Saez, & Isasi, 2018; Suganuma, Shirakawa, & Nagao, 2017), feature selection for machine learning (Kim, Street, & Menczer, 2000; Xue, Zhang, Browne, & Yao, 2015), image processing (Bochinski, Senst, & Sikora, 2017; Nickolay, Schneider, & Jacob, 1997) and so on.

The second group of bio-inspired algorithms, swarm intelligence, was inspired by the social behavior expressed by the group of otherwise simple and primitive individuals: bees, ants, moths, fireflies, dragonflies, bats, fish, etc. These individuals in swarms exhibit coordinated and highly intelligent actions, without any dedicated central unit which will organize and coordinate all other individuals. This characteristic of the swarms was used as an inspiration for swarm intelligence algorithms (Yang, 2014).

One of the first algorithms which was introduced in the domain of swarm intelligence is the particle swarm optimization algorithm (PSO) (Kennedy & Eberhart, 1995). The PSO performs the search by simulating the behavior of the flocks of birds and fish. This algorithm was successfully applied in solving numerous practical problems, including scheduling problems in the cloud computing (Kumar & Sharma, 2018). Another important representative of swarm algorithms with numerous applications for different NP-hard problems is artificial bee colony (ABC). The ABC metaheuristic has been tested against benchmark problems (Bacanin & Tuba, 2012), and had also been applied to practical NP-hard problems, as it can be seen from Kulkarni, Desai, and Kulkarni (2016), Tuba and Bacanin (2014), and Cheng, Qu, and Xu (2017).

Another well-known swarm algorithm is bat algorithm (BA) (Yang, 2010), with numerous applications in a wide range of domains, i.e. solving the workflow scheduling problem in the cloud computing (Sagnika, Bilgaiyan, & Mishra, 2018). Another popular swarm metaheuristics is cuckoo search (CS) (Gandomi, Yang, & Alavi, 2013), which has also been successfully applied to numerous problems, such as cloud computing (Agarwal & Srivastava, 2018) and neural networks training (Tuba, Alihodzic, & Bacanin, 2015). Ant colony optimization (ACO) was one of the first swarm algorithms, and it has proven to be one of the most efficient approaches, as stated in Jovanovic and Tuba (2011, 2013). Firefly algorithm (FA), inspired by the behavior of the fireflies and their lighting properties, has been extensively used in solving several NP-hard problems, in modified (Bacanin & Tuba, 2014; Strumberger, Bacanin et al., 2017) and hybridized versions (Tuba & Nebojsa, 2014).

There are also other numerous novel swarm intelligence algorithms, and important representatives are monarch butterfly optimization (MBO) and moth search algorithms (MS). MBO was initially proposed by Wang and Deb in 2015 (Wang, Deb, & Cui, 2015), and was applied to numerous practical NP-hard problems with promising results including wireless sensor networks localization problem (Strumberger, Tuba, Bacanin, Beko, & Tuba, 2018), cloud computing optimization problems (Strumberger, Tuba, Bacanin, & Tuba, 2019) and many others (Strumberger, Tuba, Bacanin, Beko, & Tuba, 2018, 2020). The MS algorithm, on the other hand, was proposed in 2016 by Wang (2016). It was inspired by the behavior of moths, more precisely phototaxis, and Lévy flights of the moths. MS has proven to be one of the best algorithms for global optimization benchmark problems, and also showed promising results for solving some real-life NP-hard problems, such as the problem of the drone placement (Strumberger, Sarac, Markovic, & Bacanin, 2018) and localization problem in the wireless sensor networks (Strumberger, Tuba, Bacanin, Beko, & Tuba, 2018). Besides already mentioned algorithms, many other swarm algorithms exist, such as elephant herding optimization (EHO) (Correia, Beko, Cruz, & Tomic, 2018; Strumberger, Bacanin, Beko, Tomic, & Tuba, 2017; Strumberger, Beko, Tuba, Minovic, & Bacanin, 2018; Strumberger, Minovic et al., 2019), tree growth algorithm (TGA) (Strumberger, Tuba, Bacanin, & Tuba, 2019; Strumberger et al., 2019), brain storm optimization (BSO) (Tuba, Strumberger, Bacanin, Zivkovic, & Tuba, 2018), and many others (Strumberger, Bacanin, Tuba, & Tuba, 2019).

As it has been already mentioned, the ANFIS has shown some very promising results for prediction model development in a wide spectrum of different domains. In order to achieve good prediction, the training process of ANFIS is crucial. Nevertheless, the quality and precision of the model can be further improved drastically, by optimizing the model parameters. There are numerous optimization methodologies available, however, the most promising approach is the application of the swarm intelligence metaheuristics to reinforce the parameters and outputs of the ANFIS. In Karaboga and Kaya (2019b), ABC algorithm was applied for ANFIS optimization, in order to estimate the number of foreign visitors coming to Turkey. The same authors proposed training ANFIS with and adaptive and hybrid ABC algorithm, as shown in Karaboga and Kaya (2019c). Another recent paper (Mir, Kamyab, Lariche, Bemani, & Baghban, 2018) discusses ANFIS paired with PSO with a goal to estimate gas density based on the pressure, temperature, molecular weight, and other important gas parameters. The proposed ANFIS-PSO model was more accurate when compared to other gas prediction models. In the manuscript (Al-qaness et al., 2020) the flower pollination algorithm (FPA) by using the salp swarm algorithm (SSA) to improve the ANFIS was proposed. SSA is applied to improve the FPA flaws, such as getting trapped in the local optima.

Hybridized ANFIS approach was utilized recently in numerous applications in the domain of sustainability. Work presented in Seifi, Ehteram, Singh, and Mosavi (2020) deals with six metaheuristics approaches used to hybridize the artificial neural network (ANN) and ANFIS with a goal to predict the monthly groundwater level. Authors were able to conclude that the approach where ANFIS was hybridized with grasshopper optimization algorithm (GOA), called ANFIS-GOA, showed superior performance and enhanced the ANFIS accuracy drastically. ANFIS-PSO approach was used in Adedeji, Akinlabi, Madushele, and Olatunji (2020) to predict the potential power output of wind turbines. The proposed ANFIS-PSO was compared to the standalone ANFIS, and provided better forecast accuracy, with a cost in a higher computational time. The research presented in Xu, Huang, Li et al. (2020) proposed the ANFIS hybridized with the vibration particle swarm optimization (VPSO). ANFIS-VPSO was then utilized to optimize the reasoning system in the milling process and reduce the energy consumption, while improving the efficiency of the tools. In Yaseen et al. (2019), authors evaluated three different algorithms, namely PSO, GA and differential evolution (DE), and integrated them with the ANFIS with a goal to predict rainfall time series. The presented results showed that all three hybridized approaches, ANFIS-PSO, ANFIS-GA and ANFIS-DE performed better than the conventional ANFIS. Hybrid ANFIS-PSO and ANFIS-DE were analyzed and compared in Dormishi, Ataei, Khaloo Kakaie, Mikaeil, and Shaffiee Haghshenas (2019) with a goal to predict and optimize the performance of gang saw in the process of cutting the carbonate rocks. The obtained results showed that the ANFIS-PSO performance were more superior than ANFIS-DE and conventional ANFIS. Hybridized ANFIS approach was also analyzed in Bemani, Baghban, and Mosavi (2020), where authors compared and evaluated ANFIS coupled with five different evolutionary algorithms for predicting the diffusivity coefficient of carbon dioxide. ANFIS was hybridized with PSO, GA, ACO, DE and backpropagation (BP) algorithms. Obtained results showed that the hybrid ANFIS-PSO outperforms all other approaches. Finally, ANFIS-VOA (virus optimization algorithm) approach was utilized in Behnood, Golafshani, and Hosseini (2020) for predicting the COVID-19 infection rate by observing various climate-related variables. Different hybridized ANFIS implementations for various research problems are shown in Table 1.

3. Overview of adaptive neuro-fuzzy inference system

Neuro-fuzzy systems are widely used today to model various real-life problems. They have gained popularity among the scientific society because they efficiently combine the advantages of fuzzy logic and artificial neural networks. The fuzzy logic component takes care of the learning abilities, while the artificial neural network component takes the feature interpretation from fuzzy logic. By using these two approaches together, it is possible to eliminate the drawbacks of individual components, and neuro-fuzzy systems have proven to have much more superior features. The ANFIS, which was originally developed by

Table 1
Hybridized ANFIS implementations.

Method	Problem description	Reference
ANFIS-ABC	Estimation of tourists coming to Turkey	Karaboga and Kaya (2019b)
ANFIS-PSO	Estimation of gas density based on gas parameters	Mir et al. (2018)
ANFIS-FPASSA	Forecasting COVID19 cases in China	Al-qaness et al. (2020)
ANFIS-GOA, CSO, WA, GA, KA, PSO	Forecasting the monthly groundwater level	Seifi et al. (2020)
ANFIS-PSO	Forecasting the output of wind turbines	Adedeji et al. (2020)
ANFIS-VPSO	Optimization of the tools for milling process	Xu, Huang, Li et al. (2020)
ANFIS-PSO, GA, DE	Forecasting the rainfall time series	Yaseen et al. (2019)
ANFIS-PSO, DE	Optimizing the gang saw rock cutting	Dormishi et al. (2019)
ANFIS-PSO, GA, ACO, DE, BP	Predicting the diffusivity of carbon dioxide	Bemani et al. (2020)
ANFIS-VOA	Predicting COVID19 infections rate	Behnood et al. (2020)

Jang in 1993 (Jang, 1993), belongs to the group of neuro-fuzzy systems. It is based on the Takagi–Sugeno inference model (Angelov & Filev, 2004; Johansen, Shorten, & Murray-Smith, 2000), which generates the mappings between the inputs and the outputs by obtaining and applying IF-THEN rules. To achieve this goal, the ANFIS model has to be trained.

The error is given by the difference obtained when comparing the output during the training with the actual output of the observed system. Based on the error status, the parameters of the ANFIS model are repeatedly updated to achieve the optimum structure of the model. Fig. 1 shows one example of an ANFIS structure, which consists of two inputs and one output, and five layers in total. In fact, the neural network architecture which is utilized in ANFIS consists of five fixed layers: fuzzification (layer one), fuzzy inference system (layers two and three), defuzzification (layer four), and aggregation (layer five).

On layer one, every node is adaptive with one parametric activation function. Membership functions use values of the inputs to obtain fuzzy clusters. Different membership functions can be utilized to calculate the membership values, where some of the most commonly used functions include generalized bell function, trapezium, triangle, gaussian, and sigmoid. These calculated membership values are within the range of $[0,1]$. Parameters a_i, b_i, c_i are used to set the form of the utilized membership function, and they are used in ANFIS training. These parameters are often referred to as antecedent parameters. The output is the membership degree of input values which satisfy the membership functions. For example, generalized bell membership function is given with Eqs. (1) and (2).

$$\mu_{A_i} = gbellmf(x;a,b,c) = \frac{1}{1+|\frac{x-c}{a}|^{2b}} \tag{1}$$

$$O_i^1 = \mu_{A_i}(x) \tag{2}$$

On layer two, each node is a fixed node, and output is the product of the input signal. Typically, it applies the fuzzy operation AND. Firing strengths w_i are calculated by utilizing the membership values which are the output of level one. Values w_i are obtained as a product of the membership values, represented in Eq. (3):

$$O_i^2 = w_i = \mu_{A_i}(x) \times \mu_{B_i}(y), i = 1,2 \tag{3}$$

On layer three, each node is fixed, and it computes the normalized firing strengths for each rule by utilizing the firing strengths which are the output from level two. Normalized firing strength for the rule i is computed as a ratio of firing strength of the rule i relative to the sum of all firing strengths, represented in Eq. (4):

$$O_i^3 = \overline{w_i} = \frac{w_i}{w_1 + w_2}, i = 1,2 \tag{4}$$

On layer four, which is known as a defuzzification layer, each node is adaptive. Here, the output for each rule is calculated by multiplying the normalized firing strength from the previous layer by a first-order polynomial. The set of polynomial's parameters $\{p_i, q_i, r_i\}$ are known as the conclusion parameters, which are used in the training of the ANFIS model. The output for every rule is computed by using Eq. (5):

$$O_i^4 = \overline{w_i} f_i = \overline{w_i}(p_i x + q_i y + r_i) \tag{5}$$

On level five, every node is fixed and adds all incoming values. Therefore, the final output of ANFIS is a sum of outputs of each rule from the level four, and it can be computed using Eq. (6):

$$O_i^5 = \sum_i \overline{w_i} f_i = \frac{\sum_i w_i f_i}{\sum_i w_i} \tag{6}$$

The training process of ANFIS in practice refers to the optimization of the parameters used in the model. ANFIS parameters include the number of inputs to the system, types, and the number of the membership function utilized in the model, and a total number of rules used in the model. Together with the antecedent and conclusion parameters, this represents a set of parameters that can be optimized. In this paper, an enhanced BAS algorithm has been employed to perform the optimization, however, only antecedent and conclusion parameters have been taken into the account, while a generalized bell membership function has been chosen. Therefore, the type of membership function was not subject to the optimization process.

4. Proposed hybrid machine learning method

In this section, the method that was proposed in this research will be described. First, an overview of the original BAS will be outlined. Then, insights into basic BAS's deficiencies will be discussed and the improved BAS algorithm that overcomes those deficiencies will be explained. Lastly, a developed ANFIS-based framework that utilizes enhanced BAS metaheuristics for the training will be shown.

4.1. Original BAS algorithm

The BAS algorithm is a novel bionic algorithm introduced in 2017 by the Jiang and Li (2017). The algorithm was inspired by the behavior of longhorn beetles, more precisely by the process of detecting and searching for food. The longhorn beetles have two antennas which they use to detect the food smell concentration. If the higher smell concentration is detected by the left antennae, the longhorn beetle will fly to the left. Similarly, if the smell concentration detected by the right antennae is higher, it will fly to the right. By doing so, the beetle is able to find the food successfully in an unknown environment. The BAS algorithm mimics this process and it can achieve efficient optimization, without prior knowledge about the particular form of the function and its gradient. It also requires only one individual, which has a great impact on lowering the computational complexity of the algorithm. This algorithm can be utilized to enhance the calculation efficiency for the back propagation (BP) algorithm in neural networks and help it find the global optimal solution with a higher probability, by determining the hyperparameters in an intelligent manner.

This relatively novel metaheuristic has already shown some promising results on real-life optimization problems. It was applied in Xu, Huang, and Ma (2020) to improve the BP neural network model to predict the gas explosion pressures. It was also used to solve other optimization problems as well, such as path planning for mobile robots with collision-free capability (Wu, Lin, Jin, Chen, Li, & Chen, 2020), conditioning optimization of extreme learning machine (Zhang et al., 2018),

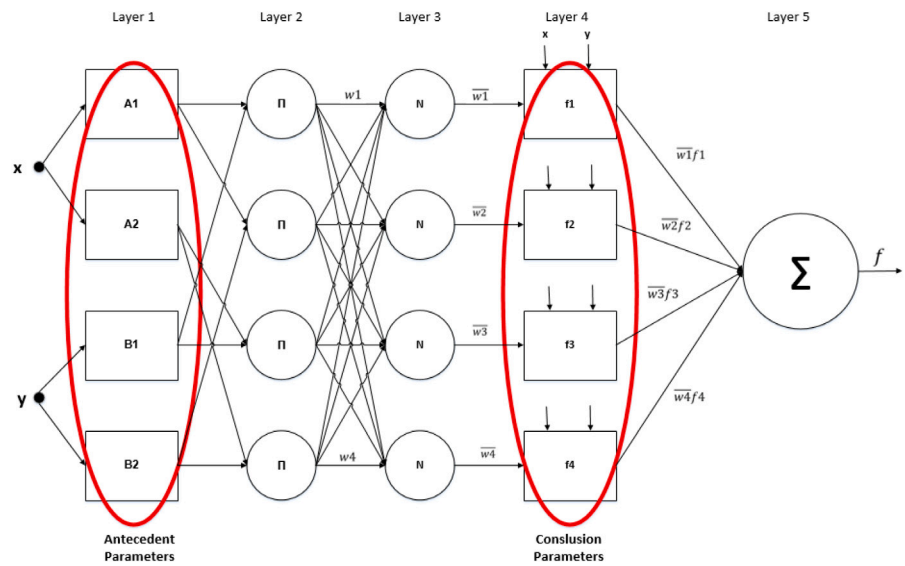

Fig. 1. ANFIS model and parameters used for training.

intelligent fault diagnosis of wind turbine rolling bearings (Wang, Yao, Cai, & Zhang, 2020) etc.

The BAS algorithm considers the position of the beetle as a vector x^t at time instant ($t = 1, 2, \ldots$) and defines the concentration of odor at position x by the fitness function $f(x)$. The maximum value of the fitness function $f(x)$ marks the source of the odor. Next, BAS algorithm utilizes two rules inspired by the beetle using antennae to search and explore an unknown environment in a random fashion. First, the searching behavior of beetle in a random direction can be modeled by Eq. (7):

$$\vec{b} = \frac{rnd(k,1)}{\| rnd(k,1) \|} \tag{7}$$

Where, rnd stands for the random function, while k represents the dimension of the position. Afterwards, the searching behaviors of the right and left antenna respectively can be modeled by Eqs. (8) and (9):

$$x_r = x^t + d^t\vec{b} \tag{8}$$

$$x_l = x^t - d^t\vec{b} \tag{9}$$

where x_r and x_l mark the positions located on the right and left side of the searching area, respectively. The sensing range of the antenna is marked with d, and it corresponds to the exploit ability, which must be large enough to cover an adequate searching area in order to be capable of jumping out of local minimum points at the beginning and then attenuate as time elapses.

The detecting behavior is formulated by the iterative model, which associates the detection of odor by considering the searching behavior, as described by Eq. (10).

$$x^t = x^{t-1} + \delta\vec{b}sign(f(x_r) - f(x_l)) \tag{10}$$

where δ represents the step size of each iteration, and $sign()$ is the sign function. The searching parameters, such as antenna length d and step size δ, are updated according to the rules given by Eqs. (11) and (12):

$$d^t = 0.95d^{t-1} + 0.01 \tag{11}$$

$$\delta^t = 0.95\delta^{t-1} \tag{12}$$

The pseudo-code of the original BAS is presented in Algorithm 1 below.

Algorithm 1 Pseudo-code of the original BAS algorithm

Input: Objective function $f(x^t)$ is established, with variables $x^t = [x_1, ..., x_i]^T$, and initialization of parameters x^0, d^0, δ^0
Output: x_{bst}, f_{bst}
while $t < T_{max}$ or ($stopcriterion$) **do**
  Generate direction vector $\vec{b}$ by using Eq. (7);
  Search in variable space with antennae by using Eq. (8) and Eq. (9);
  Update x^t by using Eq. (10);
  if $f(x^t) < f_{bst}$ **then**
    $f_{bst} = f(x^t), x_{bst} = x^t$
    Update diameter d and step size δ with decreasing functions given by Eq. (11) and (12), respectively.
  end if
end while
Return x_{bst}, f_{bst}

4.2. Improved BAS algorithm

Similarly as in Wu et al. (2020) and Xie, Chu, Zheng, and Liu (2019), by running empirical simulations, it can be concluded that some components of the original BAS metaheuristics could be enhanced. The main drawbacks of the basic BAS refer to the premature convergence with implications that the search process may be trapped in local optimums. In some runs, the diversity of the population in early iterations is not on the satisfying level and the whole population, due to the stochastic nature, may converge to sub-optimal solutions. This scenario arises when initial pseudo-random individuals are deployed in parts of the search space that are far from optimum regions.

The original BAS algorithm in each iteration performs search around the current solution x (Eqs. (8)–(10)). Exploration and exploitation processes, as well as the balance between them, are controlled by antenna lengths (d) and the step size (δ). Those are also parameters that will be updated in each iteration by using Eqs. (11) and (12) for d and δ, respectively. However, by adjusting the values of these parameters it is very hard to establish an appropriate balance between exploration and exploitation, which is in most cases adjusted in favor of exploitation because the search process is guided by the position of the current solution x. The simulations with 500 runs have been performed for standard unconstrained benchmark instances, and it was evident that on average 25% the original BAS could not converge properly and that leads to the unacceptable mean values. Therefore it can clearly be concluded that the original BAS could be improved by establishing stronger exploration in early iterations. At this stage of the algorithm's

execution, it is important to have stronger diversification, so the algorithm could find part of the search region, where the optimum solution resides.

To efficiently address observed deficiencies of the basic BAS, two mechanisms have been incorporated:

1. inspired by the approach presented in C., G., H., and T. (2019), Cauchy mutation operator has been adopted in the original BAS to improve solutions diversity and
2. to control whether Cauchy perturbation (mutation) operator will be executed, or not, a mechanism that is similar to one used in the ABC algorithm has been implemented (Karaboga & Akay, 2011)

First, the goal is to improve exploration ability and solutions diversity of original BAS in early phases of execution by incorporating Cauchy mutation operator. The Cauchy distribution is utilized to conduct Cauchy variation on solutions that do not converge in cis consecutive iterations (cis is additional control parameter of improved BAS which will be explained later). The basic idea behind this approach is that some solutions may be trapped in the local extreme hence an external intervention is required (Cao, Iosifidis, Chen, & Gabbouj, 2018), so the search can be redirected towards exploration (global search) (El-Ela, El-Sehiemy, & Abbas, 2018).

For single dimension random variable $C(0,\sigma)$ of Cauchy distribution, density function is defined as C. et al. (2019):

$$f_c(x) = \frac{\sigma}{[\pi(\sigma^2 + x^2)]}, \ -\inf < x < +\inf, \tag{13}$$

where $\sigma = 1$ represents standard Cauchy distribution.

In many evolutionary algorithms' implementations, Cauchy and Gaussian mutations operators are used. However, since the peak of Cauchy distribution at the origin is smaller than in Gaussian and the speed of converging towards axis indefinitely at both ends is slower, Cauchy distribution is more efficient in generating random numbers, which can be substantial in avoiding algorithm to fall into sub-optimum domains (C. et al., 2019). In the proposed improved BAS implementation, for each parameter j of solution x, a step vector of length d (in this case d represents number of solutions' parameters) is generated in the following way:

$$step_j = \begin{cases} (ub_j - lb_j) \cdot C(0,1) & \text{if } rnd \le CMP, \\ 0 & \text{otherwise} \end{cases} \tag{14}$$

where ub_j and lb_j denote the maximum and minimum value of parameter j, respectively, CMP is the Cauchy mutation probability, while the Cauchy variation expression is represented as $C(0,1)$. New solution in t iteration x^t_{new} is then generated by using Eq. (15):

$$x^t_{new} = x^{t-1} + step \tag{15}$$

The global exploration ability of Cauchy operator is needed in early iterations of the algorithm's run. However, in the later phases, when the search has converged to the optimal domain, this mechanism is not useful anymore. To control this behavior, a new parameter has been added — Cauchy mutation invocation (cmi). If the condition $t \le cmi$ is satisfied, then the Cauchy mutation is triggered, otherwise, the new solution x is created as in the original BAS algorithm, according to Eq. (10).

Moreover, in early phases of algorithms execution, Cauchy mutation operator will be applied (Eq. (15)) only to solutions that are not being improved in cis consecutive iterations. The consecutive iterations stagnation (cis) is another control parameter of the improved BAS algorithm. To incorporate this behavior, each solution in the population is encoded by using attribute — not improved counter (nic). In each iteration, if the solution is not improved by using the standard BAS equation, the nic is incremented by one. Finally, when conditions $nic = cis$ and $t \le cmi$ are met, Cauchy operator will be triggered. As it can be seen from Eq. (14), only for certain parameters, when condition $rnd \le CMP$ is satisfied, Cauchy variation expression is applied. This method allows for greater control over the global search. If this operator would be applied to all parameters, the exploration will be too strong, and the balance will be set in favor of a global search that would generate lower solutions' quality.

Moreover, in order to establish better adjustable balance between exploitation and exploration, attenuation coefficient (α) has been employed for the step size δ and the antenna length (sensing diameter) d along with the minimum antenna length (sensing diameter length) d_0. Similar approach was performed in Xie et al. (2019) and Wu et al. (2020). Expression of the original BAS for calculating these two parameters (Eqs. (11) and (12)) are replaced with the following ones:

$$d^t = \alpha d^{t-1} + d_0 \tag{16}$$

$$\delta^t = \alpha \delta^{t-1} \tag{17}$$

Motivated by the nature of modifications, the proposed approach is named Cauchy exploration strategy BAS (CESBAS). By introducing Cauchy mutation and three additional control parameters (cis, cmi and CMP) proposed improved CESBAS outperforms basic BAS versions, as it is shown in Section 5. It must be noted here that the optimal values of the parameters which were used in this manuscript were determined empirically, by conducting simulations with trial and error approach.

Pseudo-code of the proposed CESBAS metaheuristics is given in Algorithm 2. In the provided pseudo-code, n represents total number of solutions in the population, f denotes the fitness function, x^{it} marks ith solution in the tth iteration, while x_{bst} and f_bst denote representation of the best solution and its fitness, respectively.

Algorithm 2 Pseudo-code of the CESBAS metaheuristics

```
Initialize random population P = [x^1, x^2, x^3, ..., x^n], calculate f for all
solutions and initialize parameters d^0 and δ^0
while t < T_max or (stop criteria) do
  for all i = 1 to n do
    if nic = cis and t ≤ cmi then
      Update x^it by applying Cauchy mutation Eq. (15)
    else
      Generate direction vector b⃗ by using Eq. (7)
      Search in variable space with antennae by using Eqs. (8) and
      (9)
      Update x^it by using Eq. (10)
    end if
    Calculate f(x^it)
  end for
  Update diameter d and step size δ with decreasing functions given
  by Eqs. (16) and (17), respectively.
  Update x_bst and f_bst
end while
Return x_bst, f_bst
```

4.3. Proposed hybrid method

Since the ANFIS parameters have a significant influence on the overall ANFIS system performance, and an optimum combination of parameters' values represent NP-hard optimization problem, swarm algorithms could be applied to improve ANFIS time series forecasting. The goal of the proposed hybrid method is to enhance ANFIS performance by determining its parameters via CESBAS metaheuristics approach. The hybrid approach was named CESBAS-ANFIS.

The process of training ANFIS refers to the optimization of its structure and parameters for a specific problem. The number of inputs and rules, along with type and number of membership functions provide the total number of parameters in the ANFIS structure. In the proposed

Table 2
Benchmark function details.

ID	Function name	Function definition	Global minimum	Search domain
f1	Ackley	$f(x) = -20e^{(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2})} - -e^{(\frac{1}{n}\sum_{i=1}^{n}\cos(2\pi x_i))} + 20 + e^{(1)}$	$f(0,0) = 0$	$-5 \le x, y \le 5$
f2	Rastrigin	$f(x) = \sum_{i=1}^{n}[x_i^2 - 10\cos(2\pi x_i) + 10]$	$f(0,\dots,0) = 0$	$-5.12 \le x_i \le 5.12$
f3	Sum Squares	$f(x) = \sum_{n=1}^{N}(nx_n^2)$	$f(0) = 0$	$x_i \in [-10, 10]$
f4	Sphere	$f(x) = \sum_{n=1}^{N} x_n^2$	$f(0,\dots,0) = 0$	$-\infty \le x_i \le \infty, 1 \le i \le n1 \le i \le n$
f5	Griewank	$f(x) = 1 + \sum_{i=1}^{n}\frac{x_i^2}{4000} - \prod_{i=1}^{n} cos(\frac{x_i}{\sqrt{i}})$	$f(0,\dots,0) = 0$	$x_i \in [-600, 600]$
f6	Salomon	$f(x) = 1 - cos(2\pi\sqrt{\sum_{i=1}^{D} x_i^2}) + 0.1\sqrt{\sum_{i=1}^{D} x_i^2}$	$f(0,\dots,0) = 0$	$x_i \in [-100, 100]$

approach, the total number of parameters that should be optimized is represented as the sum of antecedent and conclusion parameters. In devised hybrid CESBAS-ANFIS method, a similar strategy has been used as in Al-qaness et al. (2020). The proposed method is based on the classic ANFIS model and employs five layers (Fig. 1). Input variables are provided in Layer 1, while Layer 5 generates foretasted values. The best weights between layers 4 and 5 are determined by the CESBAS approach in the ANFIS training process.

At the beginning of execution, the CESBAS-ANFIS prepares input data by formatting it in time series form. As in Al-qaness et al. (2020), for this purpose, autocorrelation function (ACF) has been used, as means to find the patterns in the data. Variables with ACF value greater than 0.2 have been considered. To train and evaluate the model, the train-test-split approach has been used, with the 75% of data set used for training, while the remaining 25% was used for testing. Moreover, the fuzzy c-mean (FCM) method was used for ANFIS model construction. The ANFIS parameters are then trained by the CESBAS metaheuristics. The best solution (ANFIS structure) generated by the CESBAS is then returned to the ANFIS and the test phase is performed with this solution. Each CESBAS solution represents one ANFIS structure. The length of each solution is the sum of antecedent and conclusion parameters. The type of membership function was not considered.

To calculate fitness of each solution s (potential ANFIS structure) in the training phase by the CESBAS, the mean square error (MSE) metrics is used:

$$MSE_s = \frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i - y_i)^2 \tag{18}$$

where $\hat{y}_i$ and y_i represent the predicted and the actual data for each observation, respectively, and the total number of observations is denoted as N. The fitness of each solution s from the population (fit_s), is then calculated by utilizing the following expression:

$$fit_s = \frac{1}{MSE_s} \tag{19}$$

It should be noted here that only generalized bell function has been considered as membership function in the conducted simulations since as it was mentioned earlier the membership function was not considered as a variable of the optimization process. Flow chart diagram of the proposed CESBAS-ANFIS is shown in Fig. 2.

5. Experimental setup and simulations

The experimental section is divided into two parts. In the first part, results obtained on standard tests for unconstrained (bound-constrained) benchmarks have been shown and analyzed with the goal of validating proposed CESBAS on a wider range of benchmark instances. Since the original BAS was also tested on these benchmarks, a comparative analysis with the original BAS has been performed, as well as with one other enhanced BAS implementation and one improved PSO approach, for which the results were retrieved from the modern literature (Xie et al., 2019).

In the second part of the simulation section, results for predicting COVID-19 cases will be shown on one practical study and a comparative analysis is performed with other approaches that were tested on the same datasets and in the same experimental environment (Al-qaness et al., 2020). BAS and CESBAS approaches have been implemented in the Python environment. For testing ANFIS-BAS and ANFIS-CESBAS hybrid methods, anfis 0.3.1 module for Python has been utilized. Moreover, for the purpose of results' visualization, data science Python libraries have been utilized: scipy, pandas, pyplot and seaborn. Since each fitness function evaluation requires training and testing ANFIS with the available dataset, it utilizes a lot of computational resources. Therefore all simulations were performed on computer platform with 6 NVIDIA GTX 1080 GPUs with Intel® CoreTM i7-8700K CPU and 32 GB of RAM running under Windows 10 x64 operating system.

5.1. Simulation results for unconstrained benchmarks

Before validating CESBAS on the practical problem of COVID-19 outbreak prediction, experiments have been conducted on six well-known unconstrained benchmark instances with 20 dimensions. Formulations of benchmark functions (dataset) that were utilized in the simulations are given in Table 2.

Since BAS belongs to the group of relatively novel metaheuristics, in the literature survey of recent computer science literacy, authors have identified only one paper that provides simulation results of original BAS on standard unconstrained benchmarks (Xie et al., 2019), therefore for comparative analysis purposes the same benchmarks have been used that were utilized in this paper. Moreover, for the sake of objective comparative analysis, simulations were performed under the same conditions as in Xie et al. (2019). In this manuscript, an improved BAS was proposed and both original and improved versions were tested in 50 independent runs with 200 iterations per run and with only one solution in the population. Improved BAS was compared with the original BAS, as well as with linear decreasing weight PSO (LDWPSO). The LDWPSO algorithm was tested with 3 solutions in the populations since in the original and improved BAS three functions evaluations are performed in each iteration for each solution (left and right antenna and centroid). For more details regarding specific parameters' setup of original BAS, improved BAS, and LDWPSO, please refer to Xie et al. (2019).

Specific parameters' of the CESBAS were set as follows: α and d_0 were set to 0.95 and 0.01, respectively, CMP was set to 0.5, cis to 5, while the value for cmi was adjusted to 80. The values of all parameters were determined empirically by conducting simulations. It must be pointed out that the same values for α and d_0 were used as in the original BAS (Eqs. (11) and (12)) (Jiang & Li, 2017). Simulation environment parameters of CESBAS approach for unconstrained benchmarks are summarized in Table 3.

In order to better present the search history and how the CESBAS algorithm performs the search, 2D Gaussian KDE (Kernel density estimation) and Surface plot of the Gaussian 2D KDE have been generated for all six unconstrained benchmark functions after 100 iterations. 2D Gaussian KDE are given in Fig. 3, while the Surface plot is shown in Fig. 4

The results for quality and convergence speed in terms of the number of iterations were taken as criteria for comparison with mean and standard deviation metrics calculated over 50 independent runs. Results for basic BAS improved BAS and LDWPSO were retrieved

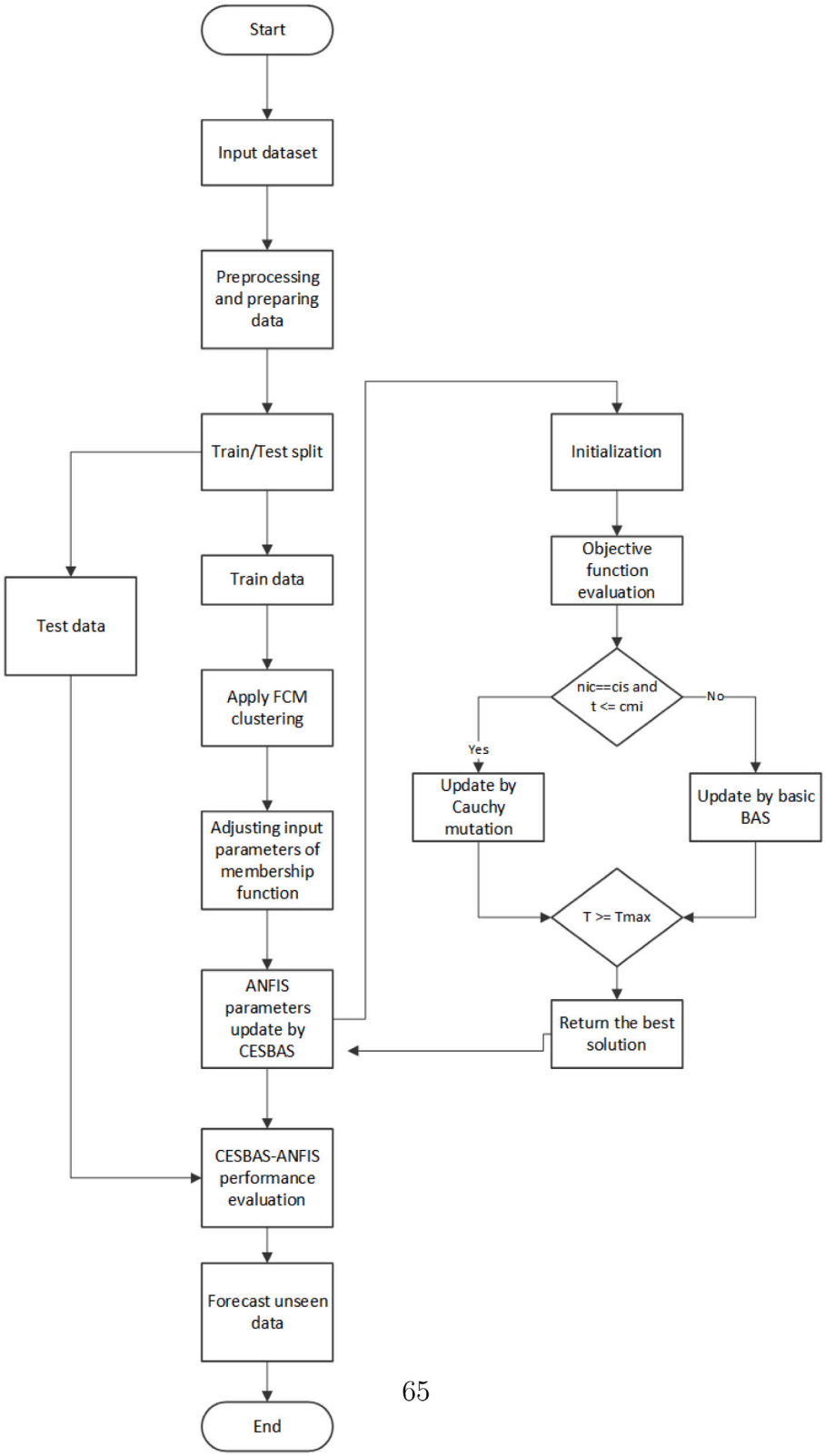

Fig. 2. CESBAS-ANFIS flow chart diagram.

from Xie et al. (2019). Comparative analysis results are summarized in Table 4 (for the sake of better visual representation, the best results for each test category are marked bold).

On average, when all criteria and metrics are taken into account, the proposed CESBAS exhibits better performance than all other approaches as evident by the comparative analysis. The second best

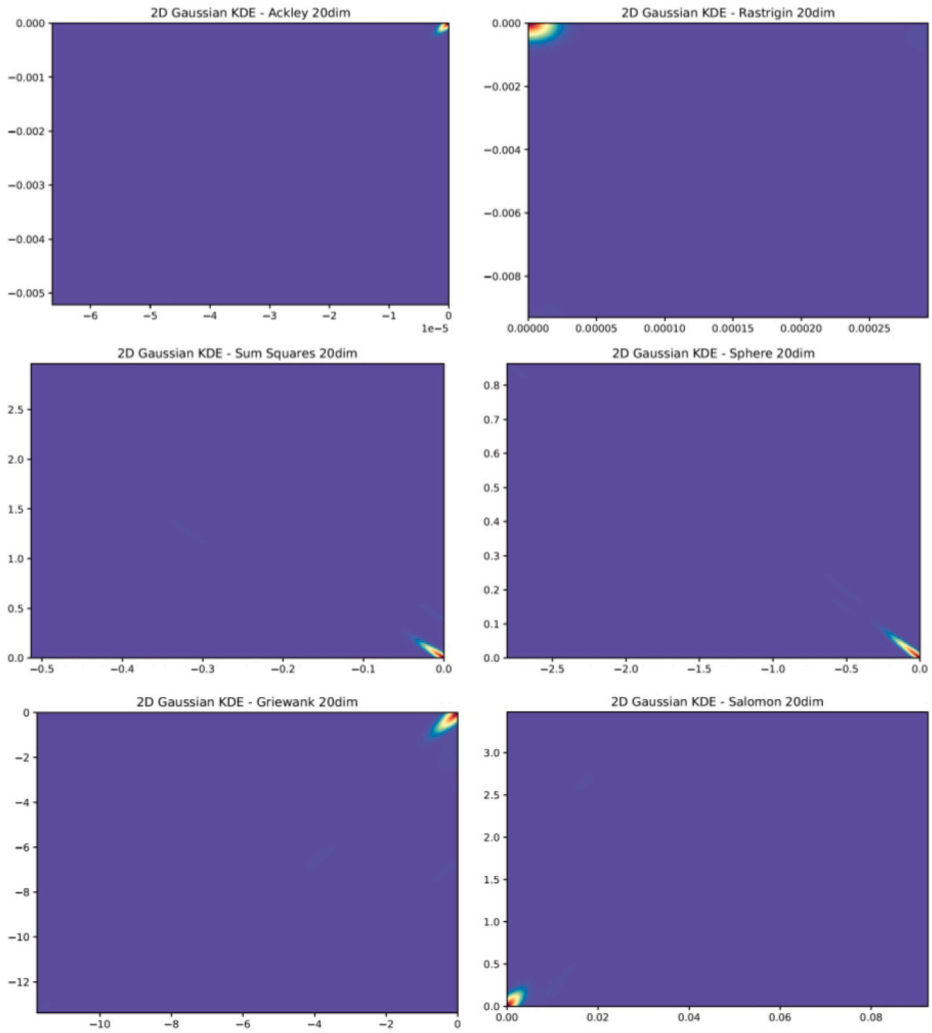

Fig. 3. CESBAS 2D Gaussian Kernel.

Table 3
CESBAS experiment parameters for unconstrained benchmarks tests.

Parameter	Notation	Value
Number of runs	$runs$	50
Number of iterations	T_{max}	200
Attenuation coefficient	α	0.95
Min sensing diameter length	d_0	0.01
Cauchy mutation probability	CMP	0.5
Cauchy mutation invocation	cmi	80
Consecutive iterations stagnation	cis	5

Table 4
Experimental results and comparative analysis for simulations with unconstrained benchmarks.

Criterion	Metrics	Metaheuristics	f1	f2	f3	f4	f5	f6
Results quality	Mean	CESBAS	**0.31**	**0.03**	0.00	0.00	**0.11**	**0.11**
		Imp.BAS	0.55	0.06	0.00	0.00	0.14	0.14
		BAS	13.16	61.21	15.18	1.37	5.47	7.05
		LDWPSO	12.74	58.24	55.84	11.31	36.70	6.19
	Std	CESBAS	**0.20**	**0.02**	0.00	0.00	**0.07**	0.10
		Imp.BAS	0.76	0.10	0.00	0.00	0.20	0.10
		BAS	6.52	23.08	17.64	2.61	12.35	2.75
		LDWPSO	5.99	23.13	70.52	12.91	39.92	4.7
Convergence	Mean	CESBAS	**47.21**	**15.13**	4.00	4.25	**3.00**	**7.15**
		Imp.BAS	55.2	20.3	**2.9**	**3.5**	3.2	10.5
		BAS	112.4	98.80	39.00	39.90	37.20	43.60
		LDWPSO	82.20	79.20	37.20	44.20	51.40	64.60
	Std	CESBAS	**23.20**	**6.20**	2.45	2.10	**1.50**	**7.00**
		Imp.BAS	38.40	13.70	**1.10**	**1.70**	1.70	8.00
		BAS	29.90	29.60	9.70	8.50	9.90	9.60
		LDWPSO	41.10	47.60	24.00	27.30	26.10	26.80

algorithm is improved BAS, which was proposed in Xie et al. (2019). In results quality comparison, four out of six tests, CESBAS outperforms improved BAS. For relatively easy benchmarks, $f3$ and $f4$, both algorithms in all runs managed to obtain global optimum. Moreover, in $f1$, $f2$ and $f5$ instances CESBAS also managed to establish better standard deviation of results than improved BAS, while in $f3$, $f4$ and $f6$ tests, metaheuristics showed the same performance.

As noted before, in Table 4, convergence time was shown in terms of the number of iterations that the algorithms took to converge to the best solution in each run. In the case of simpler functions $f3$ and $f4$, improved BAS exhibits better convergences speed than the proposed CESBAS for both indicators, mean and standard deviation. However, for all remaining functions, CESBAS managed to outperform the improved BAS. The LDWPSO was proven to be the worst approach that was considered in comparative analysis, while both CESBAS and improved BAS showed significantly better results than the basic BAS metaheuristics. It can be concluded from the comparative analysis table that, due to the lack of exploration, in some runs BAS could not

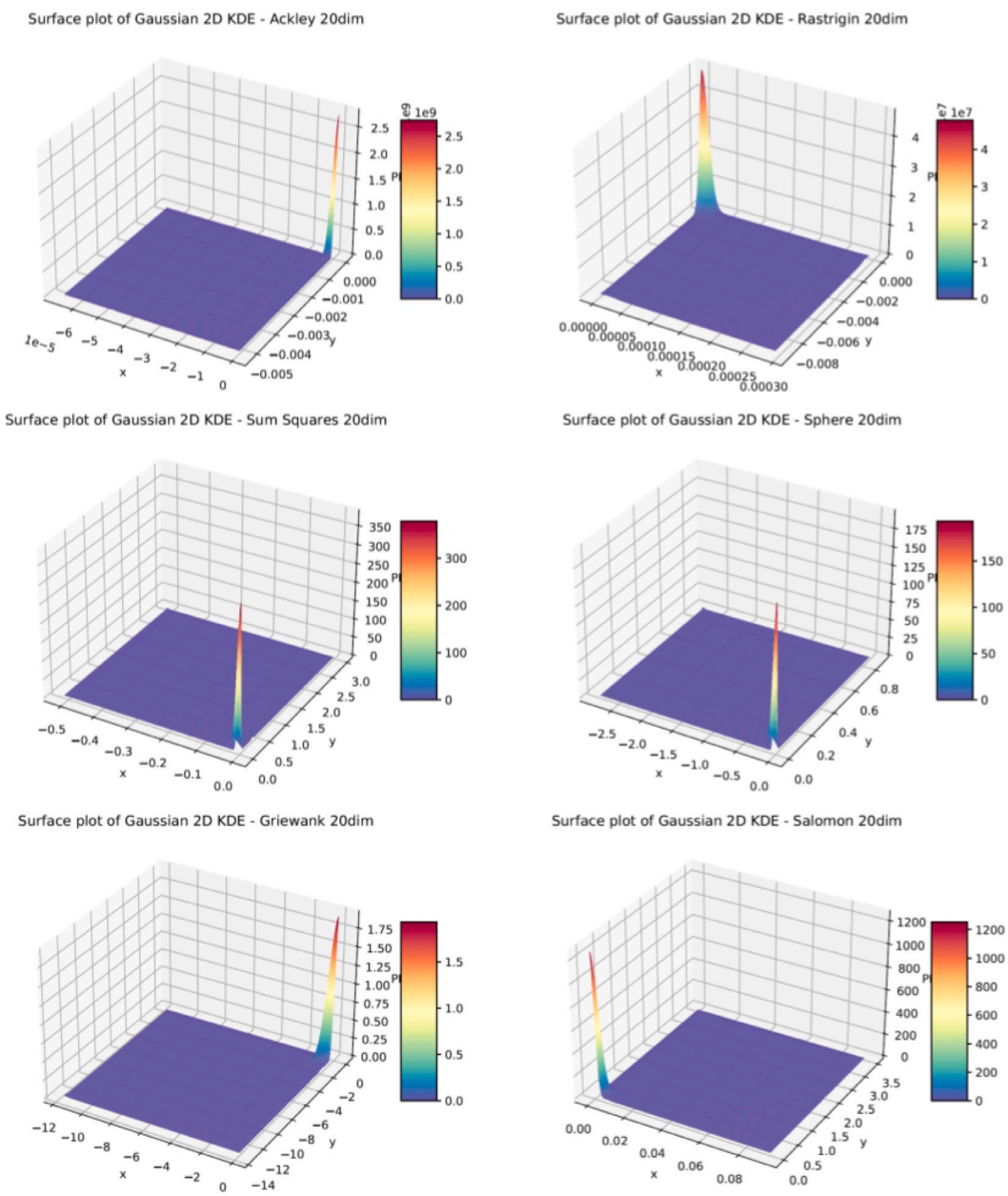

Fig. 4. CESBAS 2D Gaussian Kernel Surface plot.

achieve the right part of the search space which led to worse mean values. Even for simple $f3$ and $f4$ benchmarks, BAS failed to achieve optimum. The same can be stated for convergence speed. The proposed improved CESBAS managed to significantly improve results quality as well as convergence speed of the original BAS algorithm and obtained a significantly better results in all tests for all criteria and performance metrics.

Visual comparative analysis of results quality between CESBAS and the basic BAS is given in Figs. 5–7. Swarm plot diagrams of the best obtained results for 50 independent runs is shown in Fig. 5. Each point in the diagram represents the result of one run. Similarly, in Figs. 6 and 7, box plot diagrams (box and whiskers) and histogram are shown, respectively from the same data set. From the presented figures it is clear that in some runs, basic BAS did not converge to the optimum region, and these results are distributed far away from the median value.

The visual representation and comparative analysis between the proposed CESBAS and the original BAS was performed while taking into account convergence speed criteria and is presented in Fig. 8. On the given graphs, convergence speed averaged over 50 independent runs is shown. From the given figure it is obvious that CESBAS shows much better convergence than original BAS metaheuristics.

5.2. COVID-19 cases prediction simulations

As noted above, in the second part of simulations, CESBAS was validated against an important and current challenge of predicting COVID-19 cases by using the dataset from China. In this subsection, performance metrics that are used for testing the proposed CESBAS-ANFIS method are shown first. Then, the employed dataset and control parameter setup are shown, and finally comparative analysis with other state-of-the-art approaches that were tested on the same dataset and under similar experimental conditions is presented. For more details about the proposed CESBAS-ANFIS hybrid approach, please refer to Section 4.3.

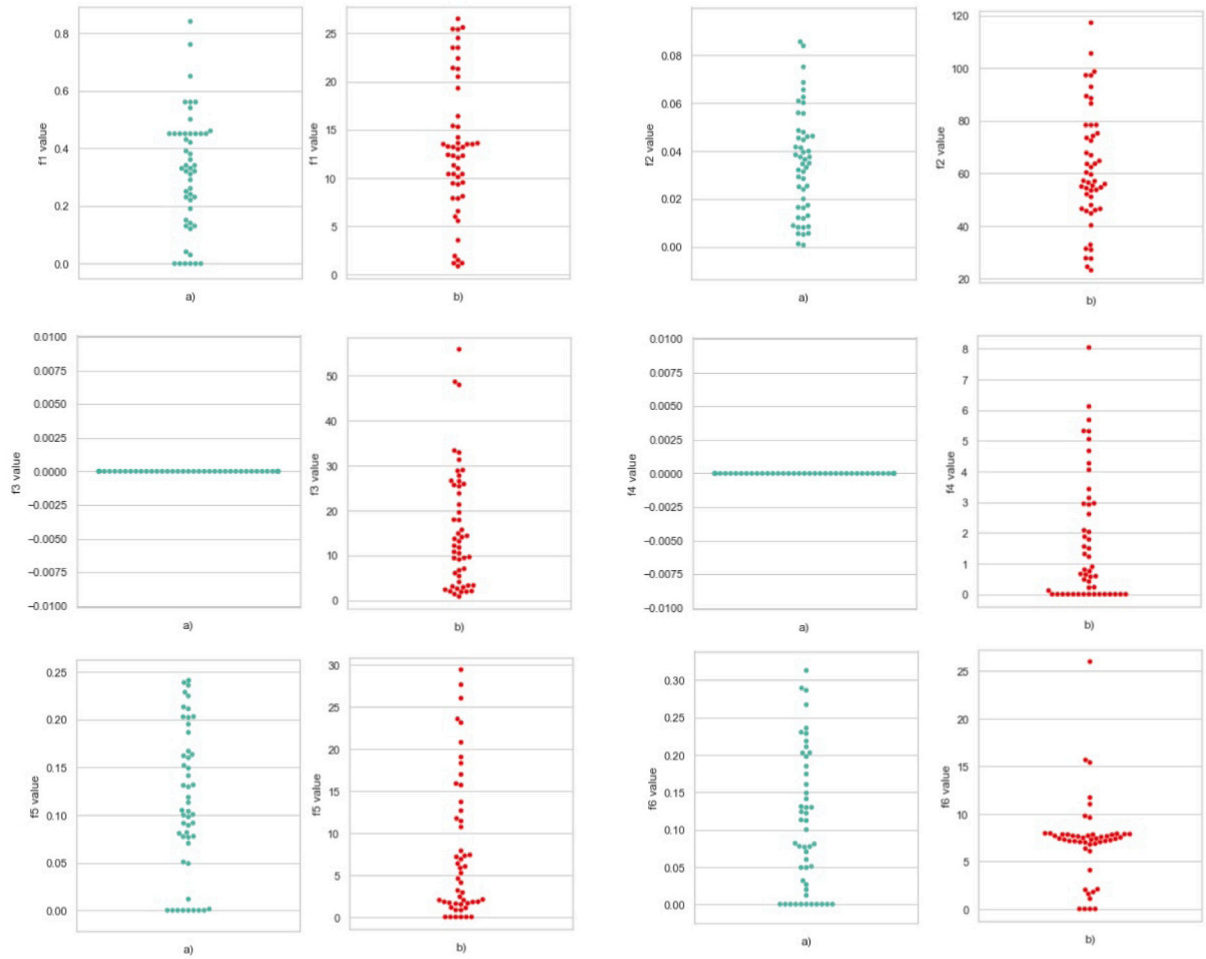

Fig. 5. Swarmplots - (a) CESBAS vs. (b) BAS.

5.2.1. Performance metrics

The quality and performance of the proposed CESBAS-ANFIS approach have been evaluated by utilizing standard metrics for regression: root mean square error (RMSE), mean absolute error (MAE), mean absolute percentage error (MAPE), root mean squared relative error (RMSRE) and coefficient of determination (R^2). Mathematical expressions of these metrics are provided in the following few paragraphs.

The RMSE is calculated as given in Eq. (20).

$$RMSE = \sqrt{\frac{1}{N_s}\sum_{i=1}^{N_s}(\hat{y}_i - y_i)^2} \tag{20}$$

where $\hat{y}_i$ and y_i stand for the predicted and actual values, respectively, and the data sample size (number of observation) is given with the parameter N_s.

The MAE, MAPE, RMSRE and (R^2) indicators are calculated by Eq. (21), Eq. (22), Eq. (23) and (24), respectively:

$$MAE = \frac{1}{N_s}\sum_{i=1}^{N_s}|\hat{y}_i - y_i| \tag{21}$$

$$MAPE = \frac{1}{N_s}\sum_{i=1}^{N_s}\frac{|\hat{y}_i - y_i|}{\hat{y}_i} \tag{22}$$

$$RMSRE = \sqrt{\frac{1}{N_s}\sum_{i=1}^{N_s}\left(\frac{\hat{y}_i - y_i}{\hat{y}_i}\right)^2} \tag{23}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y}_i)^2} \tag{24}$$

where $\bar{y}$ denotes the average value of y. Smaller values of RMSE, RMSRE, MAE, and MAPE indicate better performance of the proposed approach, while the higher value of R^2 indicates a better correlation and thus better results' quality.

5.2.2. Datasets and control parameters' setup

To test the performance of the proposed method, certain time period of COVID-19 dataset from China was used. Moreover, the performance of the proposed ANFIS-CESBAS was also compared with the hybrid between flower pollination algorithm and salp swarm algorithm (FPASSA) which was originally tested for the same problem (Al-qaness et al., 2020). In this paper, the FPASSA was also used for updating ANFIS parameters (FPASSA-ANFIS). Additionally, a comparative analysis was conducted with other state-of-the-art swarm algorithms used for updating ANFIS parameters as well as with the original BAS algorithm.

For the sake of performing all of the aforementioned comparisons, in this experiment, the same dataset was taken and simulated in a similar experimental conditions as in Al-qaness et al. (2020). It can be noted that a larger dataset could have been taken, however, in this scenario it would not be possible to compare the performance of the proposed method with other algorithms, due to the fact that only a few methods, which results are published in the state-of-the-art journals, were implemented and evaluated for COVID-19 cases prediction. Since the implementation of the original BAS algorithm for this problem does not exist in the literature, to evaluate improvements of the CESBAS over original BAS for training ANFIS parameters, during this research, BAS-ANFIS has also been implemented.

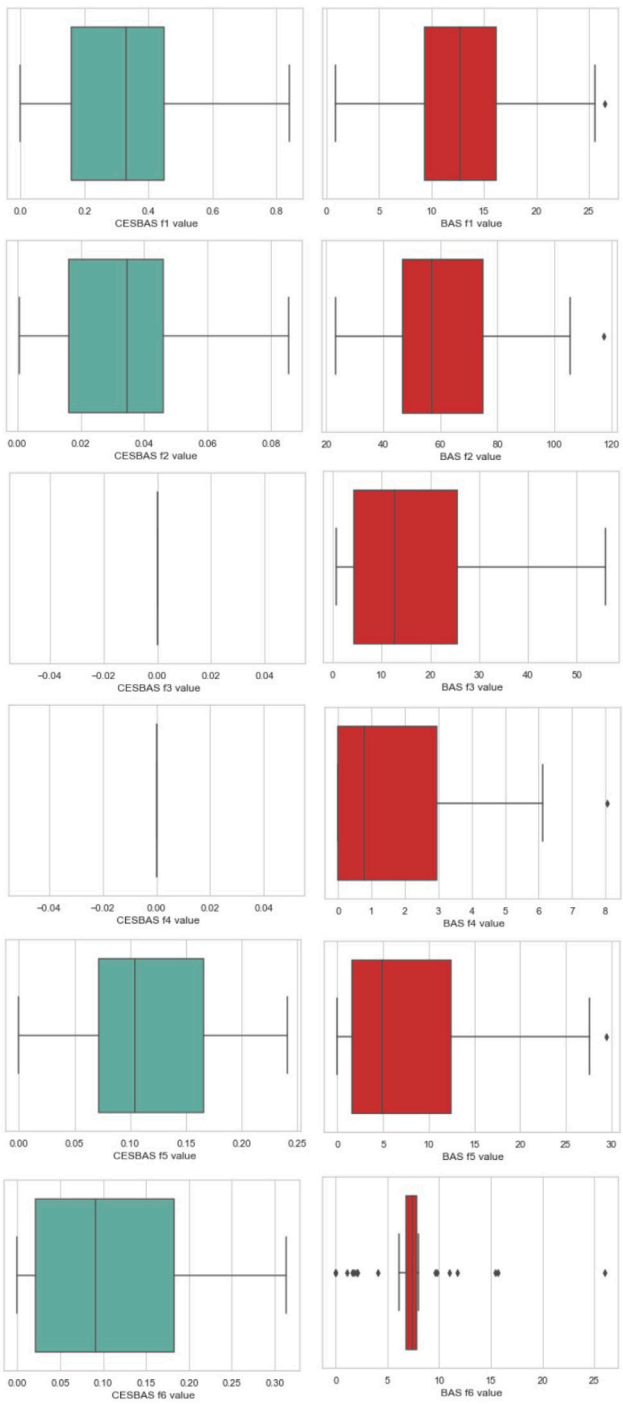

Fig. 6. Box plot diagrams CESBAS vs. BAS.

The dataset that was employed in experiments was retrieved from the World Health Organization (WHO) by merging reports of daily confirmed cases in China from January 21, 2020, till February 18, 2020. The data was captured from the following URL: www.who.int/emergencies/diseases/novel-coronavirus-2019/situation-reports/. This dataset is shown in Table 5, while its visual representation is provided in Fig. 9. Also, as in Al-qaness et al. (2020), 25% of the data was utilized for testing and the remaining 75% was used for training. In order to allow readers to see more clearly how the proposed CESBAS algorithm works, and to see the discrepancies between predicted and actual cases,

Table 5
Confirmed COVID-19 cases in China from January 21, 2020 till February 18, 2020, retrieved and merged from WHO reports.

Date (D/M/Y)	Confirmed cases	Date (D/M/Y)	Confirmed cases
21/1/2020	278	5/2/2020	24363
22/1/2020	309	6/2/2020	28060
23/1/2020	571	7/2/2020	31211
24/1/2020	830	8/2/2020	34598
25/1/2020	1297	9/2/2020	37251
26/1/2020	1985	10/2/2020	40554
27/1/2020	2741	11/2/2020	42708
28/1/2020	4537	12/2/2020	44730
29/1/2020	5997	13/2/2020	46550
30/1/2020	7736	14/2/2020	48548
31/1/2020	9720	15/2/2020	50054
1/2/2020	11821	16/2/2020	51174
2/2/2020	14411	17/2/2020	70635
3/2/2020	17283	18/2/2020	72528
4/2/2020	20471		

Table 6
Confirmed COVID-19 cases in China from January 21, 2020 till February 18, 2020, retrieved and merged from WHO reports, along with the predicted cases by CESBAS method.

Date (D/M/Y)	Confirmed	Predicted	Date (D/M/Y)	Confirmed	Predicted
21/1/2020	278	261	5/2/2020	24363	24701
22/1/2020	309	275	6/2/2020	28060	28798
23/1/2020	571	523	7/2/2020	31211	31760
24/1/2020	830	771	8/2/2020	34598	35090
25/1/2020	1297	1369	9/2/2020	37251	37802
26/1/2020	1985	2020	10/2/2020	40554	41550
27/1/2020	2741	2597	11/2/2020	42708	43802
28/1/2020	4537	4563	12/2/2020	44730	45090
29/1/2020	5997	6133	13/2/2020	46550	46100
30/1/2020	7736	7902	14/2/2020	48548	47610
31/1/2020	9720	9570	15/2/2020	50054	51200
1/2/2020	11821	11520	16/2/2020	51174	54950
2/2/2020	14411	14490	17/2/2020	70635	57320
3/2/2020	17283	17120	18/2/2020	72528	61423
4/2/2020	20471	20190			

the results which CESBAS obtained are shown along with the actual confirmed cases in Table 6.

It can be noticed from Table 6 that the proposed algorithm was not able to predict the large surge of the new cases between 16.2.2020 and 17.2.2020, leading to a slightly worse metrics. This is expected behavior, as in any machine learning algorithm for predicting the time series there is an error due to external unpredictable factors. The accuracy of the prediction depends on two factors, reducible and irreducible error, as shown in Eq. (25):

$$E(Y-\hat{Y})^2 = [f(X)-\hat{f}(X)]^2 + Var(\epsilon) \tag{25}$$

where $E(Y-\hat{Y})^2$ denotes the average of the squared distance between the actual and predicted value of Y, $[f(X)-\hat{f}(X)]^2$ represents the reducible error, while $Var(\epsilon)$ denotes the irreducible error. Irreducible error cannot be reduced, no matter how well the prediction is performed (James, Witten, Hastie, & Tibshirani, 2014). The reason for this is that the prediction depends on some unmeasured variables which are useful, but unknown.

In order to establish a better analysis of CESBAS-ANFIS performance, additional simulations have been conducted by using two datasets of confirmed influenza cases on a weekly basis, as in Al-qaness et al. (2020). The data for the first dataset (influenza dataset 1 - IDS1) was retrieved from the Center for Disease Control and Prevention (CDS) and this data refers to the time period between fourteenth week in 2015 and the sixth week in 2020 (Center for Disease Control and Prevention (CDS), 2020). The second dataset (influenza dataset 2 - IDS2), which was captured from the WHO website, and it comprises the data of confirmed influenza cases in China from week one in 2016 until the week 8 in 2020 (World Health Organization (WHO), 2020).

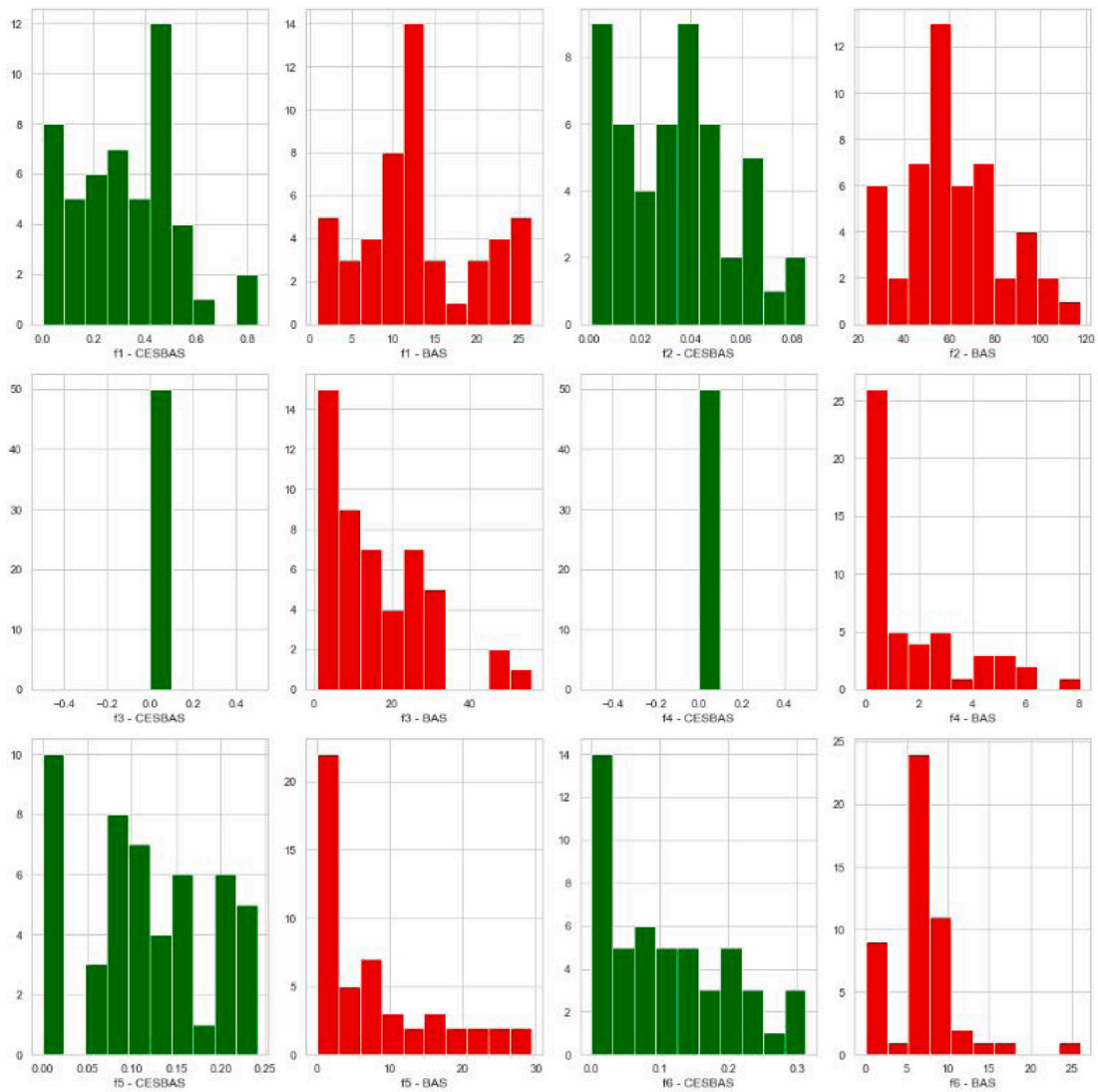

Fig. 7. Results' quality histogram CESBAS vs. BAS.

Global CESBAS-ANFIS and BAS-ANFIS control parameters were adjusted similar to what was done in Al-qaness et al. (2020): population size was set to 8 and the number of iterations was adjusted to 100. Algorithms that were implemented in Al-qaness et al. (2020) were tested by using 25 solutions in the population. However, as in the case of unconstrained benchmarks tests, since both BAS and CESBAS algorithms perform three function evaluations per iteration for each solution, experiments were conducted with only 8 solutions in the population. Other BAS and CESBAS control parameters were set as shown in Table 3. Algorithms are executed in 30 independent runs and best values are noted in the comparative analysis table.

In both experiments, with COVID-19 and influenza datasets, previous time series have been considered as independent variables, while the prediction of COVID-19 and influenza new cases are considered as dependent variables. Time-series dataset was prepared in the following way: time-series data is categorized into four inputs for the last four consequently even days' of confirmed cases that are used for predicting x_t, as the next day's confirmed case. There is no prediction horizon, as in every following iteration the algorithm takes the predicted values for calculating the next value, and so on. This methodology is visualized in Fig. 10.

5.2.3. *Experimental results, comparative analysis and discussion*

In Al-qaness et al. (2020), besides proposed FPASA-ANFIS, results of other bio-inspired algorithms that were also used for updating ANFIS parameters were shown: ANFIS-GA, ANFIS-PSO, ANFIS-GA and ANFIS-FPA. Additionally, standard machine learning algorithms and methods were also included in the analysis including: artificial neural network (ANN), K nearest neighborhoods (KNN), support vector regression (SVR) and bare bones ANFIS. All these approaches were included in comparative analysis along with the original BAS. Comparative analysis is given in Table 7. In the presented table, the results of the best run are recorded for each method along with the performance metrics. Comparative analysis also includes computation time, however, this metrics cannot be objectively compared since the approaches in this paper have been tested on different computation platform than algorithms shown in Al-qaness et al. (2020). It can be noted that authors in Al-qaness et al. (2020) have not provided details of the computation platform that was used in simulations.

In provided comparative analysis (Table 7) best obtained results for each performance metric were marked in bold style. Established results categorically prove that the proposed CESBAS-ANFIS substantially outperforms all other approaches included in the comparative analysis by establishing the best results for all performance indicators that were taken into consideration. Hybrid FPASSA-ANFIS approach shows relatively good performance, as the second-best method included

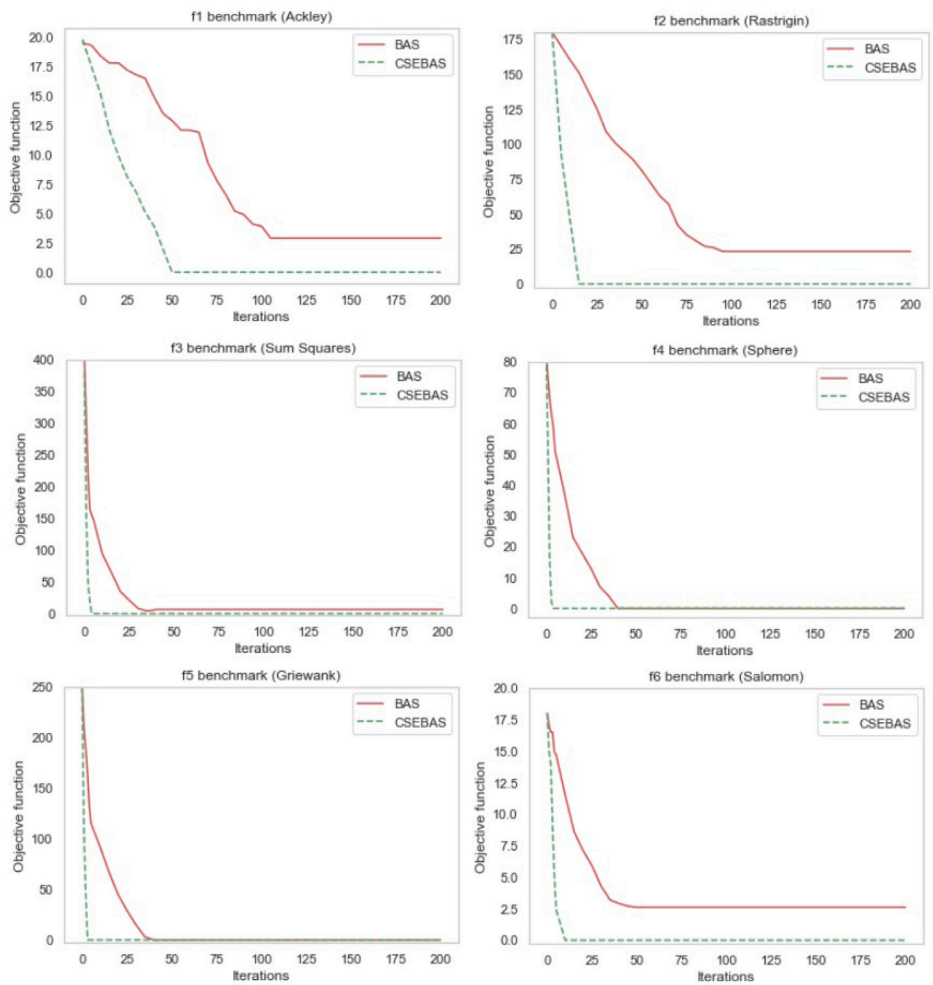

Fig. 8. Convergence speed — CESBAS vs. BAS.

Table 7
Comparative analysis between CESBAS-ANFIS and other methods for COVID-19 outbreak prediction results for China dataset retrieved from the WHO reports.

Method	RMSE	MAE	MAPE	RMSRE	R^2	time
ANN	8750	5413	13.09	0.204	0.8991	–
KNN	12100	7671	8.32	0.130	0.7710	–
SVR	7822	5354	8.40	0.080	0.8910	–
ANFIS	7375	5523	5.32	0.09	0.9032	–
PSO-ANFIS	6842	4559	5.12	0.08	0.9492	24.18
GA-ANFIS	7194	4963	5.26	0.08	0.9575	27.02
ABC-ANFIS	8327	6066	6.86	0.10	0.7906	46.80
FPA-ANFIS	6059	4379	5.04	0.07	0.9439	23.41
FPASSA-ANFIS	5779	4271	4.79	0.07	0.9645	23.30
BAS-ANFIS	7069	5125	6.56	0.10	0.7952	16.66
CESBAS-ANFIS	**4329**	**3195**	**4.08**	**0.06**	**0.9763**	19.82

in the analysis, however still significantly lower than the proposed CESBAS-ANFIS. For example, the FPASSA-ANFIS manages to obtain R^2 score of 0.9645, while CESBAS-ANFIS achieves R^2 of 0.9763, which is relatively high.

Also, from the presented results it can be seen that the basic BAS (BAS-ANFIS) shows relatively modest performance when compared to other metaheuristics and performs alike ABC. Both, BAS-ANFIS and ABC-ANFIS for all metrics obtain similar results, with the slight advantage of the BAS metaheuristics. Considering "pure" machine learning approaches (ANN, KNN and SVR), BAS-ANFIS performs better. It is also interesting to notice that the bare bones ANFIS obtains better results' quality than ABC-ANFIS and BAS-ANFIS.

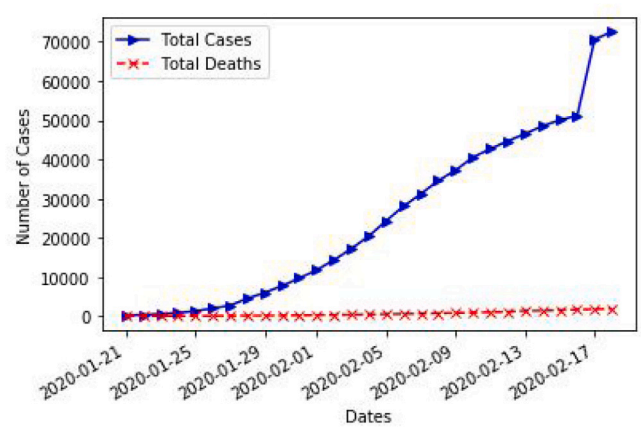

Fig. 9. Visual representation of confirmed COVID-19 cases in China from January 21, 2020 till February 18, 2020 that were retrieved from WHO reports.

5.2.3.1. CESBAS-ANFIS vs. BAS-ANFIS comparative analysis with world health organization dataset. With the goal of evaluating performance improvements of proposed CESBAS-ANFIS over the BAS-ANFIS, authors have further performed the comparative analysis of these two methods. Visual representation of obtained results for CESBAS-ANFIS and BAS-ANFIS, which is shown in Fig. 11, clearly indicate that CESBAS-ANFIS

Fig. 10. Input and output visualization of proposed CESBAS-ANFIS method.

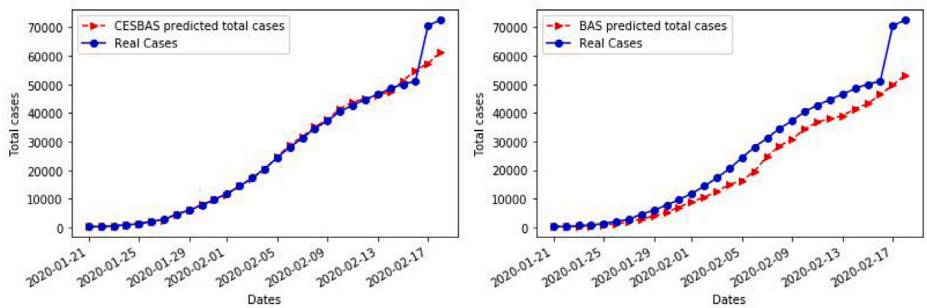

Fig. 11. Prediction of confirmed COVID-19 cases in China from January 21, 2020 till February 18, 2020 based on the data retrieved from WHO reports — CESBAS (left) and BAS (right).

manages to predict total cases in China with significantly greater accuracy than BAS-ANFIS.

Since both approaches were executed in 30 independent runs, with the goal of more detailed comparative analysis, results have been ranked, where the run with the rank 1 obtains best results, and run with the rank 30 obtains the worst result. Based on this data, a visual comparative analysis has been generated between CESBAS-ANFIS and BAS-ANFIS for RMSE and MAE indicators by using bar charts. This comparison is given in Fig. 12. Moreover, to perform detailed comparative analysis and to see results' quality distribution of 30 runs, swarm plot diagrams have also been generated of the best results obtained in each run for RMSE and MAE metrics. This analysis is provided in Fig. 13.

From the presented visual swarm plot comparative analysis it can be clearly seen that in some runs, BAS-ANFIS (labeled with b) in the figure misses the right part of the search space those results can be considered as outliers. In this example, in five runs, the BAS-ANFIS completely underscores and generates results with low quality (high RMSE and MAE values). This behavior is a consequence of the bad exploitation–exploration trade-off. However, CESBAS-ANFIS in all runs manages to generate satisfying results and there is no single run, where it misses the right part of the search space. In conclusion, it is stated that also in this practical example can be seen that the CESBAS overcomes drawbacks of the original BAS algorithm.

5.2.3.2. COVID-19 cases prediction in China with Our World in Data dataset for the periods January 21, 2020 - February 18, 2020 and November 10, 2020–December 10, 2020. When analyzing COVID-19 data that is available on the Internet, one more reliable source has been found on the web site Our World in Data (URL: https://ourworldindata.org/coronavirus-source-data). It can be observed that there are slight discrepancies between data from this web site and the data provided by the WHO reports for the observed period of time in China. In the WHO data, there has been a substantial increase in the number of new cases between February 12, 2020 and February 13, 2020 (from 44k to 59k). However, according to Our World in Data, even larger new cases increase happened between February 16, 2020 and February 17, 2020 (from 51k to 70k). This could potentially have an influence on the accuracy of prediction hence the authors decided to test CESBAS-ANFIS and BAS-ANFIS by using this dataset as well. This approach was taken to evaluate the robustness of CESBAS-ANFIS and BAS-ANFIS frameworks. This method can also reveal if significant changes in performance metric values would be seen. This dataset is visually represented in Fig. 14

Table 8
COVID-19 outbreak results for dataset retrieved from Our World in Data website.

Method	RMSE	MAE	MAPE	RMSRE	R2	time
BAS-ANFIS	6864	4960	6.53	0.10	0.7954	16.50
CESBAS-ANFIS	**4106**	**2994**	**4.08**	**0.06**	**0.9775**	19.43

Experiments have been performed under the same conditions as that was used for the WHO data and the results are presented for CESBAS-ANFIS and BAS-ANFIS in Table 8.

Results presented in Tables 7 (data from the WHO) and Table 8 (data from the website Our World in Data) are just slightly different, which is excepted. Therefore, it can be concluded that both methods, CESBAS-ANFIS and BAS-ANFIS are not susceptible to changes in datasets. Visual representation of obtained results for CESBAS-ANFIS and BAS for the dataset retrieved from the Our World in Data are shown in Fig. 15.

Finally, the authors have taken recent data related to confirmed cases in China from the *Our World in Data* source, from the period of previous month (from October 10, 2020 till November 9, 2020) and trained proposed CESBAS-ANFIS model. The goal was to try to predict number of potential cases in the following thirty days time period (from November 10, 2020 until December 10, 2020). Results of the prediction are shown in Table 9.

5.2.3.3. The infection rate trends and the climate variables. In order to further evaluate the proposed CESBAS-ANFIS method, additional experiment was conducted by including the climate-related variables. This experiment was conducted by utilizing the same experimental setup as given in Behnood et al. (2020). That study utilized various climate factors to predict the speed of the COVID19 spread in the USA, while data was obtained from various sources. The observed factors included the average temperature, minimum temperature, maximum temperature, precipitation, humidity, wind speed and population density. Finally, hybridized ANFIS-VOA approach was utilized to predict the infection rate based on the aforementioned inputs. More details can be found in Behnood et al. (2020). The results of the conducted simulations are shown in the Table 10. The statistical indicators for

Fig. 12. CESBAS-ANFIS vs. BAS-ANFIS visual comparative analysis in form of bar plots for confirmed COVID-19 cases in China from January 21, 2020 till February 18, 2020, based on the data retrieved from WHO reports for RMSE (left) and MAE (right) metrics.

Table 9
Expected number of confirmed cases in China for the period November 10, 2020–December 10, 2020.

Date	Expected cases	New	Date	Expected cases	New
10/11/2020	91578	22	26/11/2020	91979	26
11/11/2020	91597	19	27/11/2020	92006	27
12/11/2020	91618	21	28/11/2020	92037	31
13/11/2020	91642	24	29/11/2020	92070	33
14/11/2020	91671	29	30/11/2020	92100	30
15/11/2020	91703	32	01/12/2020	92135	35
16/11/2020	91738	35	02/12/2020	92176	41
17/11/2020	91767	29	03/12/2020	92214	38
18/11/2020	91794	27	04/12/2020	92248	34
19/11/2020	91818	24	05/12/2020	92279	31
20/11/2020	91845	27	06/12/2020	92307	28
21/11/2020	91868	23	07/12/2020	92339	32
22/11/2020	91890	22	08/12/2020	92370	31
23/11/2020	91909	19	09/12/2020	92404	34
24/11/2020	91930	21	10/12/2020	92442	38
25/11/2020	91953	23			

linear regression, stand-alone ANFIS, ANFIS-VOA-I and ANFIS-VOA-II were obtained from the referenced paper (Behnood et al., 2020). From the presented results, it can be seen that ANFIS-BAS slightly outperforms the ANFIS-VOA-I approach, while it is still behind the ANFIS-VOA-II method. The proposed ANFIS-CESBAS method, however, manages to slightly outperform ANFIS-VOA-II method, and all other evaluated methods.

The infection rate trends with the changing of the most important climate input variables are given in Fig. 16. It can be observed that infection rate rises drastically with the increase of the population density, which could justify the need for the social distancing. It is also notable that the infection rate drops with the increase of the average temperature, while it also slightly drops with the increase of the wind. Finally, infection rate shows trends of increasing with the increase of the humidity.

5.2.3.4. Simulations with influenza datasets. Finally, the comparative analysis has been performed with approaches presented in Al-qaness et al. (2020) for confirmed influenza datasets. The description of this dataset, as well as control parameters used in simulations, are given in Section 5.2.2.

Based on the results presented in Table 11, it can be stated that when average results are taken into account, the proposed CESBAS-ANFIS establishes a better performance than all other approaches that have been included in the analysis. In simulations with IDS1 dataset, only FPASSA-ANFIS managed to obtain better MAPE value than CESBAS-ANFIS, while both approaches perform the same in terms of R^2 metric comparison.

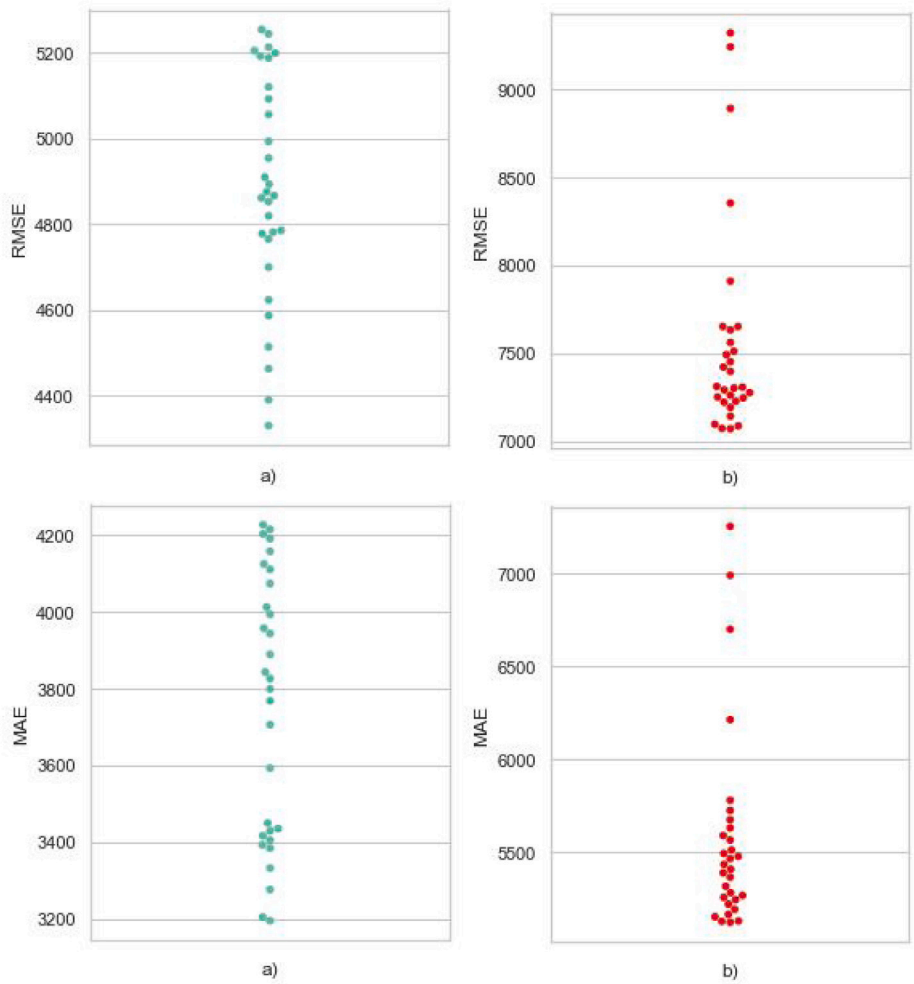

Fig. 13. CESBAS-ANFIS vs. BAS-ANFIS visual swarm plot comparative analysis of confirmed COVID-19 cases in China from January 21, 2020 till February 18, 2020, based on the data retrieved from WHO reports - (a) CESBAS, (b) BAS.

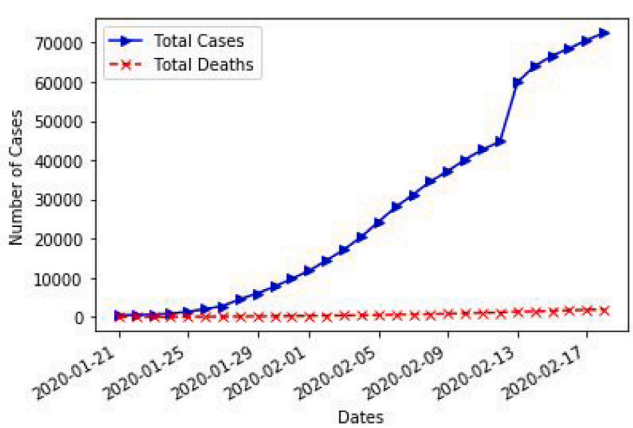

Fig. 14. Visual representation of confirmed COVID-19 cases in China from January 21, 2020 till February 18, 2020 that were retrieved from website Our World in Data.

However, the second best approach in simulations with IDS2 dataset proved to be PSO-ANFIS that managed to outperform CESBAS-ANFIS in MAE and RMSRE performance indicators. For other metrics, CESBAS-ANFIS showed better results.

Similarly, as in the case of COVID-19 prediction simulations, original BAS exhibited similar performance as the ABC. Also, as in the previous test, the proposed CESBAS managed to completely outperform original BAS by establishing a better balance between intensification and diversification.

6. Conclusion

In this manuscript, a novel method has been proposed to predict new COVID-19 cases by employing hybridized algorithm between machine learning, adaptive neuro-fuzzy inference system (ANFIS) and enhanced beetle antennae search (BAS) swarm intelligence metaheuristics. Since one of the greatest challenges in any machine learning approach is parameters' optimization and adjustments for a specific practical problem, enhanced BAS algorithm was utilized for solving this task. The proposed Cauchy exploration strategy BAS (CESBAS) was tested on a standard set of unconstrained benchmarks and proved to be a robust metaheuristics that significantly outscored all other approaches including the original BAS.

Additionally, CESBAS algorithm has been incorporated for updating ANFIS parameters and it was tested on a practical COVID-19 new cases prediction. The proposed method has been tested on the COVID-19 case study because it is currently the most important challenge the entire humanity is facing. However, the method can be generalized and applied to predict any time-series. Simulation results and comparative analysis showed that the proposed hybrid method has outperformed other sophisticated approaches that were tested on the same datasets and proved to be an useful tool for time-series prediction.

Table 10
Predicted infections rate based on the climate variables.

Method	RMSD(Infected people/Days)	MAE (Infected people/Days)	R^2	R-value
Linear regression	43.0204	12.1912	0.3925	0.6257
ANFIS	30.6515	9.0127	0.6911	0.8314
ANFIS-VOA-I	27.6533	9.0494	0.7486	0.8653
ANFIS-VOA-II	22.4744	7.3337	0.3925	0.6257
ANFIS-BAS	26.5578	9.0116	0.7562	0.8741
ANFIS-CESBAS	22.1245	7.1554	0.8457	0.9185

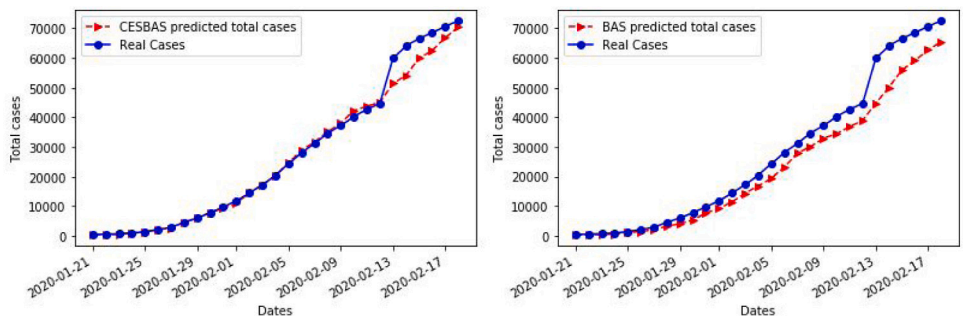

Fig. 15. Prediction of confirmed COVID-19 cases in China from January 21, 2020 till February 18, 2020 based on the data retrieved from website Our World in Data — CESBAS (left) and BAS (right).

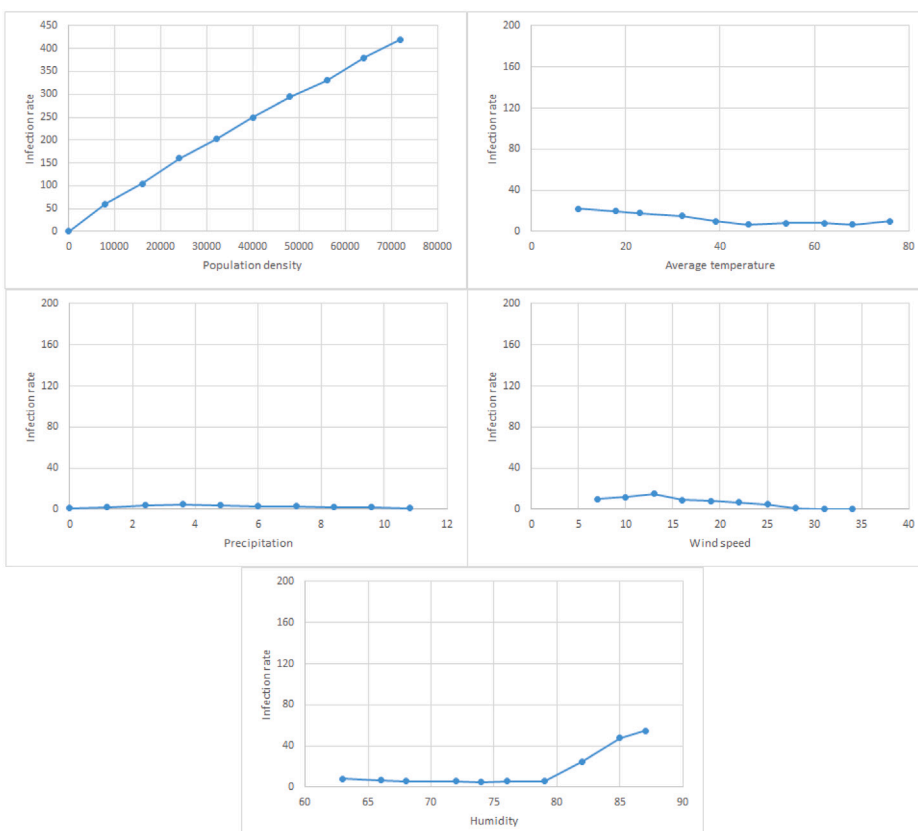

Fig. 16. Infection rate trends by different climate variables.

The primary contribution of this paper is reflected in the fact that the prediction accuracy has been enhanced for the number of new confirmed disease cases on the COVID-19 case study. The ongoing COVID-19 outbreak showed complex nature, and the promising results from this research can provide an alternative disease outbreak modeling approach, which can be used by the authorities to decide what measures should be taken, and when to implement them. The prediction accuracy of the proposed model also suggests that, in case of a disease outbreak, machine learning models can be used together with traditional epidemiological models to predict the number of new confirmed cases. Proposed method can easily be applied to any time-series prediction.

The secondary contribution of this research is the enhancement of the original BAS algorithm. Moreover, proposed CESBAS proved to be a very efficient metaheuristics that can be adapted for solving other real-life NP hard challenges.

Table 11
Computational results for datasets of confirmed influenza cases.

Dataset	Method	RMSE	MAE	MAPE	RMSRE	R^2	time
IDS1	ANFIS	952	570	37.61	0.551	0.969	–
	PSO-ANFIS	798	494	34.13	0.510	0.978	25.43
	GA-ANFIS	766	480	35.44	0.530	0.98	28.70
	ABC-ANFIS	878	564	39.79	0.593	0.972	49.27
	FPA-ANFIS	618	411	37.69	0.570	0.979	24.58
	FPASSA-ANFIS	609	391	**32.58**	0.497	0.986	24.55
	BAS-ANFIS	872	555	39.43	0.588	0.976	19.05
	CESBAS-ANFIS	**598**	**362**	32.62	**0.491**	0.986	21.32
IDS2	ANFIS	718	405	64.20	1.198	0.858	–
	PSO-ANFIS	620	**353**	52.07	**0.870**	0.892	31.64
	GA-ANFIS	622	362	87.91	3.216	0.902	34.83
	ABC-ANFIS	696	433	53.30	1.101	0.887	60.87
	FPA-ANFIS	622	371	80.55	3.152	0.898	30.42
	FPASSA-ANFIS	619	367	45.02	0.887	0.909	30.39
	BAS-ANFIS	692	435	51.15	1.529	0.879	28.29
	CESBAS-ANFIS	**618**	359	**44.91**	0.872	**0.911**	37.43

The challenge with the proposed approach is presented in the testing process. That is because every change in control parameters of the utilized metaheuristics will require a new set of simulation runs. Keeping in mind that the training of ANFIS is extremely resource intensive and time consuming process, it would be necessary to work with graphical processing units and Cuda platform to provide the interested researchers with timely results. In this version of the algorithm, everything was done locally, offline. Datasets were retrieved into the csv files, and csv files were then loaded to the python environment by utilizing the pandas library. As part of the future work it is possible to modify the current solution to be online, by utilizing a RESTful web service which would expose the appropriate endpoints, making the service available to other researchers.

It is recommended as part of future research in this domain to encompass additional modifications and improvements of the original BAS algorithm. Future work can focus on hybridizing BAS with other machine learning methods for classification, as well as for regression. Additionally, it is also possible to adapt basic and enhanced BAS versions for solving various NP hard challenges (WSN localization and energy consumption problem, cloud computing scheduling, portfolio optimization, etc.), since this metaheuristics showed a promising potential in this domain.

Declaration of competing interest

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

Acknowledgment

The paper is supported by the Ministry of Education, Science and Technological Development of Republic of Serbia, Grant No. III-44006.

References

Adedeji, P. A., Akinlabi, S., Madushele, N., & Olatunji, O. O. (2020). Wind turbine power output very short-term forecast: A comparative study of data clustering techniques in a PSO-ANFIS model. *Journal of Cleaner Production*, *254*, Article 120135.

Agarwal, M., & Srivastava, G. M. S. (2018). A cuckoo search algorithm-based task scheduling in cloud computing. In S. K. Bhatia, K. K. Mishra, S. Tiwari, & V. K. Singh (Eds.), *Advances in computer and computational sciences* (pp. 293–299). Singapore: Springer Singapore.

Al-qaness, M. A. A., Ewees, A. A., Fan, H., & Aziz, M. A. E. (2020). Optimization method for forecasting confirmed cases of COVID-19 in China. *Journal of Clinical Medicine*, *9*(3), 674. http://dx.doi.org/10.3390/jcm9030674.

Angelov, P. P., & Filev, D. P. (2004). An approach to online identification of Takagi-Sugeno fuzzy models. *IEEE Transactions on Systems, Man and Cybernetics, Part B (Cybernetics)*, *34*(1), 484–498.

Ardabili, S. F., Mosavi, A., Ghamisi, P., Ferdinand, F., Varkonyi-Koczy, A. R., Reuter, U., et al. (2020). Covid-19 outbreak prediction with machine learning. Available at SSRN 3580188.

Bacanin, N., & Tuba, M. (2012). Artificial bee colony (ABC) algorithm for constrained optimization improved with genetic operators. *Studies in Informatics and Control*, *21*(2), 137–146.

Bacanin, N., & Tuba, M. (2014). Firefly algorithm for cardinality constrained mean-variance portfolio optimization problem with entropy diversity constraint. *The Scientific World Journal*, *2014*, Article 721521, 16. Special issue Computational Intelligence and Metaheuristic Algorithms with Applications. http://dx.doi.org/10.1155/2014/721521.

Baghban, A., Jalali, A., Shafiee, M., Ahmadi, M. H., & Chau, K.-w. (2019). Developing an ANFIS-based swarm concept model for estimating the relative viscosity of nanofluids. *Engineering Applications of Computational Fluid Mechanics*, *13*(1), 26–39.

Baldominos, A., Saez, Y., & Isasi, P. (2018). Evolutionary convolutional neural networks: An application to handwriting recognition. *Neurocomputing*, *283*, 38–52. http://dx.doi.org/10.1016/j.neucom.2017.12.049, URL http://www.sciencedirect.com/science/article/pii/S0925231217319112.

Behnood, A., Golafshani, E. M., & Hosseini, S. M. (2020). Determinants of the infection rate of the COVID-19 in the US using ANFIS and virus optimization algorithm (VOA). *Chaos, Solitons & Fractals*, *139*, Article 110051.

Bemani, A., Baghban, A., & Mosavi, A. (2020). Estimating CO2-brine diffusivity using hybrid models of ANFIS and evolutionary algorithms. *Engineering Applications of Computational Fluid Mechanics*, *14*(1), 818–834.

Bibri, S. E., & Krogstie, J. (2017). Smart sustainable cities of the future: An extensive interdisciplinary literature review. *Sustainable Cities and Society*, *31*, 183–212.

Bochinski, E., Senst, T., & Sikora, T. (2017). Hyper-parameter optimization for convolutional neural network committees based on evolutionary algorithms. In *2017 IEEE international conference on image processing (ICIP)* (pp. 3924–3928). http://dx.doi.org/10.1109/ICIP.2017.8297018.

C., Z., G., Z., H., W., & T., X. (2019). Hierarchical resource scheduling method using improved cuckoo search algorithm for internet of things. *Peer-to-Peer Networking and Applications*, *12*, 1606–1614. http://dx.doi.org/10.1007/s12083-019-00801-8.

Cao, G., Iosifidis, A., Chen, K., & Gabbouj, M. (2018). Generalized multi-view embedding for visual recognition and cross-modal retrieval. *IEEE Transactions on Cybernetics*, *48*(9), 2542–2555.

Center for Disease Control and Prevention (CDS) (2020). Influenza dataset. URL https://www.cdc.gov/flu/weekly/.

Chan, J. F.-W., Yuan, S., Kok, K.-H., To, K. K.-W., Chu, H., Yang, J., et al. (2020). A familial cluster of pneumonia associated with the 2019 novel coronavirus indicating person-to-person transmission: a study of a family cluster. *The Lancet*, *395*(10223), 514–523.

Cheng, L., Qu, L., & Xu, Y. (2017). Artificial bee colony algorithm-based multiple-source localization method for wireless sensor network. In *2017 2nd IEEE international conference on computational intelligence and applications (ICCIA)* (pp. 445–448). http://dx.doi.org/10.1109/CIAPP.2017.8167257.

Coronaviridae Study Group of the International, et al. (2020). The species severe acute respiratory syndrome-related coronavirus: classifying 2019-nCoV and naming it SARS-CoV-2. *Nature Microbiology*, *5*(4), 536.

Correia, S. D., Beko, M., Cruz, L. A. D. S., & Tomic, S. (2018). Elephant herding optimization for energy-based localization. *Sensors*, *18*(9), http://dx.doi.org/10.3390/s18092849.

Dormishi, A., Ataei, M., Khaloo Kakaie, R., Mikaeil, R., & Shaffiee Haghshenas, S. (2019). Performance evaluation of gang saw using hybrid ANFIS-DE and hybrid ANFIS-PSO algorithms. *Journal of Mining and Environment*, *10*(2), 543–557.

El-Ela, A. A., El-Sehiemy, R. A., & Abbas, A. S. (2018). Optimal placement and sizing of distributed generation and capacitor banks in distribution systems using water cycle algorithm. *IEEE Systems Journal*, *12*(4), 3629–3636.

Gandomi, A. H., Yang, X.-S., & Alavi, A. H. (2013). Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems. *Engineering with Computers*, *29*(1), 17–35.

Goldberg, D. E. (1989). *Genetic algorithms in search, optimization and machine learning* (1st ed.). Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc.

Harandizadeh, H., Toufigh, M. M., & Toufigh, V. (2019). Application of improved ANFIS approaches to estimate bearing capacity of piles. *Soft Computing*, *23*(19), 9537–9549.

Ivanov, D. (2020). Predicting the impacts of epidemic outbreaks on global supply chains: A simulation-based analysis on the coronavirus outbreak (COVID-19/SARS-CoV-2) case. *Transportation Research Part E: Logistics and Transportation Review*, *136*, Article 101922.

Jaafari, A., Zenner, E. K., Panahi, M., & Shahabi, H. (2019). Hybrid artificial intelligence models based on a neuro-fuzzy system and metaheuristic optimization algorithms for spatial prediction of wildfire probability. *Agricultural and Forest Meteorology*, *266*, 198–207.

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2014). *An introduction to statistical learning: With applications in R*. Springer Publishing Company, Incorporated.

Jang, J.-S. (1993). ANFIS: adaptive-network-based fuzzy inference system. *IEEE Transactions on Systems, Man, and Cybernetics*, *23*(3), 665–685.

Jiang, X., & Li, S. (2017). BAS: beetle antennae search algorithm for optimization problems. arXiv preprint arXiv:1710.10724.

Johansen, T. A., Shorten, R., & Murray-Smith, R. (2000). On the interpretation and identification of dynamic Takagi-Sugeno fuzzy models. *IEEE Transactions on Fuzzy systems*, *8*(3), 297–313.

Jovanovic, R., & Tuba, M. (2011). An ant colony optimization algorithm with improved pheromone correction strategy for the minimum weight vertex cover problem. *Applied Soft Computing*, *11*(8), 5360–5366.

Jovanovic, R., & Tuba, M. (2013). Ant colony optimization algorithm with pheromone correction strategy for the minimum connected dominating set problem. *Computer Science and Information Systems (ComSIS)*, *10*(1), 133–149.

Karaboga, D., & Akay, B. (2011). A modified artificial bee colony (ABC) algorithm for constrained optimization problems. *Applied Soft Computing*, *11*(3), 3021–3031.

Karaboga, D., & Kaya, E. (2019a). Adaptive network based fuzzy inference system (ANFIS) training approaches: a comprehensive survey. *Artificial Intelligence Review*, *52*(4), 2263–2293.

Karaboga, D., & Kaya, E. (2019b). Estimation of number of foreign visitors with ANFIS by using ABC algorithm. *Soft Computing*, 1–13.

Karaboga, D., & Kaya, E. (2019c). Training ANFIS by using an adaptive and hybrid artificial bee colony algorithm (aABC) for the identification of nonlinear static systems. *Arabian Journal for Science and Engineering*, *44*(4), 3531–3547.

Kennedy, J., & Eberhart, R. (1995). Particle swarm optimization. In *Proceedings of the IEEE international conference on neural networks, vol. 4* (pp. 1942–1948).

Khodaei-mehr, J., Tangestanizadeh, S., Vatankhah, R., & Sharifi, M. (2018). ANFIS-Based optimal control of hepatitis C virus epidemic. *IFAC-PapersOnLine*, *51*(15), 539–544.

Kim, Y., Street, W. N., & Menczer, F. (2000). Feature selection in unsupervised learning via evolutionary search. In *Proceedings of the Sixth ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 65–369).

Kulkarni, V., Desai, V., & Kulkarni, R. V. (2016). Multistage localization in wireless sensor networks using artificial bee colony algorithm. In *2016 IEEE symposium series on computational intelligence (SSCI)* (pp. 1–8). http://dx.doi.org/10.1109/SSCI.2016.7850273.

Kumar, M., & Sharma, S. (2018). PSO-COGENT: Cost and energy efficient scheduling in cloud environment with deadline constraint. *Sustainable Computing: Informatics and Systems*, *19*, 147–164. http://dx.doi.org/10.1016/j.suscom.2018.06.002, URL http://www.sciencedirect.com/science/article/pii/S2210537918300350.

Liu, D., Clemente, L., Poirier, C., Ding, X., Chinazzi, M., Davis, J. T., et al. (2020). A machine learning methodology for real-time forecasting of the 2019-2020 COVID-19 outbreak using internet searches, news alerts, and estimates from mechanistic models. arXiv preprint arXiv:2004.04019.

Megahed, N. A., & Ghoneim, E. M. (2020). Antivirus-built environment: Lessons learned from Covid-19 pandemic. *Sustainable Cities and Society*, *61*, Article 102350.

Mir, M., Kamyab, M., Lariche, M. J., Bemani, A., & Baghban, A. (2018). Applying ANFIS-PSO algorithm as a novel accurate approach for prediction of gas density. *Petroleum Science and Technology*, *36*(12), 820–826.

Mohammed, S. H., Ahmed, M. M., Al-Mousawi, A. M., Azeez, A., et al. (2018). Seasonal behavior and forecasting trends of tuberculosis incidence in Holy Kerbala, Iraq. *International Journal of Mycobacteriology*, *7*(4), 361–367.

Morens, D. M., Taubenberger, J. K., Harvey, H. A., & Memoli, M. J. (2010). The 1918 influenza pandemic: lessons for 2009 and the future. *Critical Care Medicine*, *38*(4 Suppl), Article e10.

Naderloo, L., Alimardani, R., Omid, M., Sarmadian, F., Javadikia, P., Torabi, M. Y., et al. (2012). Application of ANFIS to predict crop yield based on different energy inputs. *Measurement*, *45*(6), 1406–1413.

Nickolay, B., Schneider, B., & Jacob, S. (1997). Parameter optimisation of an image processing system using evolutionary algorithms. In *International conference on computer analysis of images and patterns* (pp. 637–644). Springer.

Pinter, G., Felde, I., Mosavi, A., Ghamisi, P., & Gloaguen, R. (2020). COVID-19 pandemic prediction for Hungary; a hybrid machine learning approach. *A Hybrid Machine Learning Approach (May 2, 2020)*.

Rypdal, M., & Sugihara, G. (2019). Inter-outbreak stability reflects the size of the susceptible pool and forecasts magnitudes of seasonal epidemics. *Nature Communications*, *10*(1), 1–8.

Sagnika, S., Bilgaiyan, S., & Mishra, B. S. P. (2018). Workflow scheduling in cloud computing environment using bat algorithm. In A. K. Somani, S. Srivastava, A. Mundra, & S. Rawat (Eds.), *Proceedings of first international conference on smart system, innovations and computing* (pp. 149–163). Singapore: Springer Singapore.

Scarpino, S. V., & Petri, G. (2019). On the predictability of infectious disease outbreaks. *Nature Communications*, *10*(1), 1–8.

Seifi, A., Ehteram, M., Singh, V. P., & Mosavi, A. (2020). Modeling and uncertainty analysis of groundwater level using six evolutionary optimization algorithms hybridized with ANFIS, SVM, and ANN. *Sustainability*, *12*(10), 4023.

Silva, B. N., Khan, M., & Han, K. (2018). Towards sustainable smart cities: A review of trends, architectures, components, and open challenges in smart cities. *Sustainable Cities and Society*, *38*, 697–713.

Sohrabi, C., Alsafi, Z., O'Neill, N., Khan, M., Kerwan, A., Al-Jabir, A., et al. (2020). World health organization declares global emergency: A review of the 2019 novel coronavirus (COVID-19). *International Journal of Surgery*.

Spinelli, A., & Pellino, G. (2020). COVID-19 pandemic: perspectives on an unfolding crisis. *The British Journal of Surgery*.

Strumberger, I., Bacanin, N., Beko, M., Tomic, S., & Tuba, M. (2017). Static drone placement by elephant herding optimization algorithm. In *Proceedings of the 24th telecommunications forum (TELFOR)*. http://dx.doi.org/10.1109/TELFOR.2017.8249469.

Strumberger, I., Bacanin, N., & Tuba, M. (2017). Enhanced firefly algorithm for constrained numerical optimization, IEEE congress on evolutionary computation. In *Proceedings of the IEEE international congress on evolutionary computation (CEC 2017)* (pp. 2120–2127).

Strumberger, I., Bacanin, N., Tuba, M., & Tuba, E. (2019). Resource scheduling in cloud computing based on a hybridized whale optimization algorithm. *Applied Sciences*, *9*(22), 4893.

Strumberger, I., Beko, M., Tuba, M., Minovic, M., & Bacanin, N. (2018). Elephant herding optimization algorithm for wireless sensor network localization problem. In L. M. Camarinha-Matos, K. O. Adu-Kankam, & M. Julashokri (Eds.), *Technological innovation for resilient systems* (pp. 175–184). Cham: Springer International Publishing.

Strumberger, I., Minovic, M., Tuba, M., & Bacanin, N. (2019). Performance of elephant herding optimization and tree growth algorithm adapted for node localization in wireless sensor networks. *Sensors*, *19*(11), 2515. http://dx.doi.org/10.3390/s19112515.

Strumberger, I., Sarac, M., Markovic, D., & Bacanin, N. (2018). Moth search algorithm for drone placement problem. *International Journal of Computers*, *3*.

Strumberger, I., Tuba, E., Bacanin, N., Beko, M., & Tuba, M. (2018). Modified monarch butterfly optimization algorithm for RFID network planning. In *2018 6th international conference on multimedia computing and systems (ICMCS)* (pp. 1–6). http://dx.doi.org/10.1109/ICMCS.2018.8525930.

Strumberger, I., Tuba, E., Bacanin, N., Beko, M., & Tuba, M. (2018). Monarch butterfly optimization algorithm for localization in wireless sensor networks. In *2018 28th international conference radioelektronika (RADIOELEKTRONIKA)* (pp. 1–6). http://dx.doi.org/10.1109/RADIOELEK.2018.8376387.

Strumberger, I., Tuba, E., Bacanin, N., Beko, M., & Tuba, M. (2018). Wireless sensor network localization problem by hybridized moth search algorithm. In *2018 14th international wireless communications mobile computing conference (IWCMC)* (pp. 316–321). http://dx.doi.org/10.1109/IWCMC.2018.8450491.

Strumberger, I., Tuba, E., Bacanin, N., Beko, M., & Tuba, M. (2020). Modified and hybridized monarch butterfly algorithms for multi-objective optimization. In A. M. Madureira, A. Abraham, N. Gandhi, & M. L. Varela (Eds.), *Hybrid intelligent systems* (pp. 449–458). Cham: Springer International Publishing.

Strumberger, I., Tuba, M., Bacanin, N., & Tuba, E. (2019). Cloudlet scheduling by hybridized monarch butterfly optimization algorithm. *Journal of Sensor and Actuator Networks*, *8*(3), 44. http://dx.doi.org/10.3390/jsan8030044.

Strumberger, I., Tuba, E., Bacanin, N., & Tuba, M. (2019). Dynamic tree growth algorithm for load scheduling in cloud environments. In *2019 IEEE congress on evolutionary computation (CEC)* (pp. 65–72). http://dx.doi.org/10.1109/CEC.2019.8790014.

Strumberger, I., Tuba, E., Zivkovic, M., Bacanin, N., Beko, M., & Tuba, M. (2019). Dynamic search tree growth algorithm for global optimization. In L. M. Camarinha-Matos, R. Almeida, & J. Oliveira (Eds.), *Technological innovation for industry and service systems* (pp. 143–153). Cham: Springer International Publishing.

Suganuma, M., Shirakawa, S., & Nagao, T. (2017). A genetic programming approach to designing convolutional neural network architectures. In *Proceedings of the genetic and evolutionary computation conference* (pp. 497–504). New York, NY, USA: ACM, http://dx.doi.org/10.1145/3071178.3071229, URL http://doi.acm.org/10.1145/3071178.3071229.

Sun, C., & Zhai, Z. (2020). The efficacy of social distance and ventilation effectiveness in preventing COVID-19 transmission. *Sustainable Cities and Society*, *62*, Article 102390.

Suzuki, Y., & Suzuki, A. (0000). Machine learning model estimating number of COVID-19 infection cases over coming 24 days in every province of South Korea (XGBoost and MultiOutputRegressor). medRxiv.

Tuba, M., Alihodzic, A., & Bacanin, N. (2015). Cuckoo search and bat algorithm applied to training feed-forward neural networks. In *Recent advances in swarm intelligence and evolutionary computation* (pp. 139–162). Cham: Springer International Publishing, http://dx.doi.org/10.1007/978-3-319-13826-8_8.

Tuba, M., & Bacanin, N. (2014). Artificial bee colony algorithm hybridized with firefly metaheuristic for cardinality constrained mean-variance portfolio problem. *Applied Mathematics & Information Sciences*, *8*(6), 2831–2844. http://dx.doi.org/10.12785/amis/080619.

Tuba, M., Bacanin, N., & Alihodzic, A. (2015). Multilevel image thresholding by fireworks algorithm. In *2015 25th international conference radioelektronika (RADIOELEKTRONIKA)* (pp. 326–330). http://dx.doi.org/10.1109/RADIOELEK.2015.7129057.

Tuba, M., & Nebojsa (2014). Improved seeker optimization algorithm hybridized with firefly algorithm for constrained optimization problems. *Neurocomputing*, *143*, 197–207. http://dx.doi.org/10.1016/j.neucom.2014.06.006.

Tuba, E., Strumberger, I., Bacanin, N., & Tuba, M. (2018). Bare bones fireworks algorithm for capacitated p-median problem. In Y. Tan, Y. Shi, & Q. Tang (Eds.), *Advances in swarm intelligence* (pp. 283–291). Cham: Springer International Publishing.

Tuba, E., Strumberger, I., Bacanin, N., Zivkovic, D., & Tuba, M. (2018). Cooperative clustering algorithm based on brain storm optimization and K-means. In *2018 28th international conference radioelektronika (RADIOELEKTRONIKA)* (pp. 1–5). http://dx.doi.org/10.1109/RADIOELEK.2018.8376369.

Uçar, T., Karahoca, A., & Karahoca, D. (2013). Tuberculosis disease diagnosis by using adaptive neuro fuzzy inference system and rough sets. *Neural Computing and Applications*, *23*(2), 471–483.

Uyar, K., Ilhan, U., Iseri, E. I., & Ilhan, A. (2019). Forecasting measles cases in ethiopia using neuro-fuzzy systems. In *2019 3rd international symposium on multidisciplinary studies and innovative technologies (ISMSIT)* (pp. 1–5). IEEE.

Wang, G.-G. (2016). Moth search algorithm: a bio-inspired metaheuristic algorithm for global optimization problems. *Memetic Computing*, http://dx.doi.org/10.1007/s12293-016-0212-3.

Wang, G.-G., Deb, S., & Cui, Z. (2015). Monarch butterfly optimization. *Neural Computing and Applications*, 1–20. http://dx.doi.org/10.1007/s00521-015-1923-y.

Wang, T., Liu, Z., Chen, Y., Xu, Y., & Dai, X. (2014). Load balancing task scheduling based on genetic algorithm in cloud computing. In *2014 IEEE 12th international conference on dependable, autonomic and secure computing* (pp. 146–152). http://dx.doi.org/10.1109/DASC.2014.35.

Wang, Z., Yao, L., Cai, Y., & Zhang, J. (2020). Mahalanobis semi-supervised mapping and beetle antennae search based support vector machine for wind turbine rolling bearings fault diagnosis. *Renewable Energy*.

World Health Organization, et al. (2020). *Coronavirus disease 2019 (COVID-19): situation report, 72*. World Health Organization.

World Health Organization (WHO) (2020). Influenza dataset. URL https://www.who.int/influenza.

Wu, Q., Lin, H., Jin, Y., Chen, Z., Li, S., & Chen, D. (2020). A new fallback beetle antennae search algorithm for path planning of mobile robots with collision-free capability. *Soft Computing*, *24*(3), 2369–2380.

Xie, S., Chu, X., Zheng, M., & Liu, C. (2019). Ship predictive collision avoidance method based on an improved beetle antennae search algorithm. *Ocean Engineering*, *192*, Article 106542. http://dx.doi.org/10.1016/j.oceaneng.2019.106542, URL http://www.sciencedirect.com/science/article/pii/S0029801819306766.

Xu, L., Huang, C., Li, C., Wang, J., Liu, H., & Wang, X. (2020). A novel intelligent reasoning system to estimate energy consumption and optimize cutting parameters toward sustainable machining. *Journal of Cleaner Production*, Article 121160.

Xu, Y., Huang, Y., & Ma, G. (2020). A beetle antennae search improved BP neural network model for predicting multi-factor-based gas explosion pressures. *Journal of Loss Prevention in the Process Industries*, Article 104117.

Xue, B., Zhang, M., Browne, W. N., & Yao, X. (2015). A survey on evolutionary computation approaches to feature selection. *IEEE Transactions on Evolutionary Computation*, *20*(4), 606–626.

Yadav, T., & Saxena, S. K. (2020). Transmission cycle of SARS-CoV and SARS-CoV-2. In S. K. Saxena (Ed.), *Coronavirus disease 2019 (COVID-19): Epidemiology, pathogenesis, diagnosis, and therapeutics* (pp. 33–42). Singapore: Springer Singapore.

Yang, X.-S. (2010). A new metaheuristic bat-inspired algorithm. In J. R. González, D. A. Pelta, C. Cruz, G. Terrazas, & N. Krasnogor (Eds.), *Nature inspired cooperative strategies for optimization* (pp. 65–74). Berlin, Heidelberg: Springer Berlin Heidelberg, http://dx.doi.org/10.1007/978-3-642-12538-6_6.

Yang, X.-S. (2014). Swarm intelligence based algorithms: a critical analysis. *Evolutionary Intelligence*, *7*(1), 17–28.

Yaseen, Z. M., Ebtehaj, I., Kim, S., Sanikhani, H., Asadi, H., Ghareb, M. I., et al. (2019). Novel hybrid data-intelligence model for forecasting monthly rainfall with uncertainty analysis. *Water*, *11*(3), 502.

Zhan, Z., Dong, W., Lu, Y., Yang, P., Wang, Q., & Jia, P. (2019). Real-time forecasting of hand-foot-and-mouth disease outbreaks using the integrating compartment model and assimilation filtering. *Scientific Reports*, *9*(1), 1–9.

Zhan, Z.-H., Zhang, G.-Y., Ying-Lin, Gong, Y.-J., & Zhang, J. (2014). Load balance aware genetic algorithm for task scheduling in cloud computing. In G. Dick, W. N. Browne, P. Whigham, M. Zhang, L. T. Bui, H. Ishibuchi, Y. Jin, X. Li, Y. Shi, P. Singh, K. C. Tan, & K. Tang (Eds.), *Simulated evolution and learning* (pp. 644–655). Cham: Springer International Publishing.

Zhang, X., Yang, Z., Cao, F., Cao, J., Wang, M., & Cai, N. (2018). Conditioning optimization of extreme learning machine by multitask beetle antennae swarm algorithm. arXiv preprint arXiv:1811.09100.