Energy 155 (2018) 705–720

Contents lists available at ScienceDirect

Energy

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

Forecasting energy demand, wind generation and carbon dioxide emissions in Ireland using evolutionary neural networks

Karl Mason*, Jim Duggan, Enda Howley

Discipline of Information Technology, National University of Ireland, Galway, Galway, Ireland

ARTICLE INFO

Article history:
Received 9 November 2017
Received in revised form
20 March 2018
Accepted 30 April 2018
Available online 5 May 2018

Keywords:
Wind power generation
Power demand
CO_2
Forecasting
Neural networks
Covariance matrix adaptation evolutionary strategy

ABSTRACT

The ability to accurately predict future power demands, power available from renewable resources and the environmental impact of power generation is vital to the energy sector for the purposes of planning, scheduling and policy making. Machine learning techniques, neural networks in particular, have proven to be very effective methods for addressing these challenging forecasting problems. This research utilizes the powerful evolutionary optimisation algorithm, covariance matrix adaptation evolutionary strategy, as a means of training neural networks to predict short term power demand, wind power generation and carbon dioxide intensity levels in Ireland over a two month period. The network is trained over one month and then tested over the following month. A neural network trained with covariance matrix adaptation evolutionary strategy performs very competitively when compared to other state of the art prediction methods when forecasting Ireland's energy needs, providing fast convergence, more accurate predictions and robust performance. The covariance matrix adaptation evolutionary strategy trained network also gives accurate predictions when predicting multiple time steps into the future.

1. Introduction

Worldwide there is an ever increasing demand for energy due to population growth, increased living standards and industrial development. This ever increasing appetite for energy gives rise to a number of problems, namely: 1) Producing the energy necessary to meet the power demand, 2) Reducing the harmful atmospheric pollutants that result from the power generation process, 3) Incorporating renewable energy sources into the power generation process. In each of these problems facing the energy sector, it is vital to develop accurate forecasting methods.

When generating power it is crucial to be able to accurately forecast energy demands in both the short term and long term. Long term energy forecasting enables policy makers, planners and engineers to prepare for future energy needs by building and developing infrastructure to generate power. Short term energy forecasting is critical to energy production as it is vital to know how much energy will be needed in the near future so that power generators can be scheduled to meet future energy needs.

Minimizing the carbon footprint of the power generation process is very important to the energy sector in recent years. It is well known that burning fossil fuels such as coal will produce harmful atmospheric pollutants such as Sulfur Dioxide (SO_2), Nitrogen Oxide (NO_x) and Carbon Dioxide (CO_2). This is a problem as these chemical compounds directly contribute to global warming. In the 2015 Paris Climate Conference (COP21) nearly 200 countries worldwide agreed to cut green house gas emissions in the coming years. In order to reduce the production of these harmful atmospheric pollutants, it is vital to be able to predict how much of these pollutants will be produced.

Due to the harmful atmospheric effects of burning fossil fuels, many countries world wide have resorted to renewables as a source of energy, e.g. wind, wave and solar energy. The primary drawback with these renewable resources is that they do not provide consistent energy. The fact that renewables do not produce a consistent source of energy is a major hurdle that must be overcome. In order to incorporate these environmentally friendly resources into the power generation process, it is important to know in advance how much energy will be available from these sources. This further motivates the need for accurate forecasting techniques.

Recently neural networks have become a popular machine learning approach for forecasting problems. Neural networks are function approximators inspired by the brain. One of the main design considerations when implementing neural networks is how

* Corresponding author.
E-mail address: k.mason2@nuigalway.ie (K. Mason).

https://doi.org/10.1016/j.energy.2018.04.192
0360-5442/

to optimise the network weights in order to produce the desired output for a given input. Traditionally these weights have been trained using the backpropagation algorithm. In recent years however there has been a large body of research conducted that focuses on the use of evolutionary algorithms to train these neural network weights. One of the most successful evolutionary algorithm is the Covariance Matrix Adaptation Evolutionary Strategy (CMA-ES) algorithm [1]. This algorithm uses evolutionary principles such as selection and mutation and implements a convariance matrix to represent the dependencies between variables to evolve solutions to complex real valued optimisation problems. CMA-ES has never been applied to forecasting in the energy sector despite its impressive performance. It is hoped that the effectiveness of CMA-ES as an optimisation algorithm can benefit the energy sector by evolving more accurate neural network forecasting models.

The accuracy of a neural network trained using CMA-ES will be judged using Ireland's energy sector as a case study. Ireland is a moderately sized country. The population of the Republic of Ireland is approximately 4.75 million people [2]. The population of Northern Ireland is 1.85 million people [3]. This research will evaluate energy data for the entire island, the Republic and Northern Ireland. As Ireland is an island nation in the north Atlantic, a significant portion of its energy is generated from wind. In 2015, 22.8% of its energy is generated from wind farms, however the majority of its energy is generated from thermal power generators [4]. CMA-ES will be compared to a number of state of the art forecasting methods to predict: 1) Energy demand. 2) Wind power generation. 3) CO_2 intensity. This will be done using data corresponding to a two month time period.

The research presented in this paper makes the following contributions:

1. The application of CMA-ES to forecasting problems in the energy sector.
2. Predicting Ireland's power demand, CO_2 levels and wind power generation using evolutionary neural networks.
3. To compare and contrast the effectiveness of the most well established evolutionary optimisation algorithms, i.e differential evolution, particle swarm optimisation and CMA-ES, in the context of energy forecasting.
4. To evaluate how accurate the evolved networks can make predictions for multiple time steps into the future.

The outline of the paper is as follows. Sections 2, 3 and 4 gives an overview of the relevant literature in energy forecasting, neural networks and evolutionary computing respectively. The experimental methods will be explained in Section 5. Section 6 will present the experimental results. Section 7 will give a discussion of the results. Finally, Section 8 will detail what conclusions can be made and also outline some potential avenues for future research.

2. Energy forecasting

This section will give an overview of energy demand forecasting, wind power generation forecasting and CO_2 level forecasting. This will include outlining the importance and motivation of each forecasting problem along with the prominent research conducted on each problem.

2.1. Energy demand forecasting

As the world's population is continuously increasing, energy providers are faced with the task of generating energy sufficiently to meet this increased power demand. Forecasting electrical load plays a crucial role within the energy sector for problems such as power system planning and management. Load forecasting can be categorized into 3 groups: 1) Short term 2) Medium term and 3) Long term forecasting. Short term forecasting is concerned with time scales of hours (the focus of this research), medium term is concerned with time scales of months and long term is concerned with time scales of years. Each of these are important for different reasons. Accurate short term forecasting is crucial to ensure that an uninterrupted power supply is available [5]. Medium term forecasting is important as it is used to schedule maintenance and repairs of the power generation infrastructure [6]. Finally long term forecasting plays a crucial role in influencing policy and infrastructure planning [7].

Forecasting approaches can be subdivided into two groups: 1) Statistical techniques 2) Artificial intelligence (AI) techniques. As will be discussed later on, AI techniques such as neural networks and support vector machines have the advantage of making fewer assumptions about the data and are therefore more flexible. These are referred to as non-parametric methods. Examples of statistical techniques include linear regression and autoregressive integrated moving average (ARIMA). There are a number of drawbacks with the ARIMA approach [8]. It is more rigid and has lack of generality when there is a significant change in the environment [9]. It is also known that ARIMA models forecast well for an only short period of time after the model has been trained [10]. This research will train the forecast model using data corresponding to a time period of a month and then evaluate the performance of the forecast model on data corresponding to the following month. For these reasons, methods such as ARIMA will not be implemented in this research. A major area of research into forecasting load demand includes incorporating other environmental variables into the forecasting models, e.g. time of day. For predicting wind power generation this could be other meteorological information such as temperature and forecast wind speed [11]. This research will implement a purely univariate approach that solely utilizes historical time series data to make predictions.

2.2. Wind generation forecasting

Many countries are devoting resources into renewable energy infrastructure as a solution to the harmful effects that arise from burning fossil fuels. In 2015, 22.8% of Ireland's energy was generated from wind power [4]. Wind power generation poses additional challenges when compared to conventional thermal power generation. This is due to the inherent stochasticity with wind speeds. Forecasting wind power generation accurately is therefore a crucial task when incorporating energy from wind farms to meet power demand. Accurate forecasts of wind power enable better economic dispatch, unit commitment and scheduling of thermal power generators [12]. In terms of economic dispatch, in order for power generators to be scheduled optimally to meet the power demand, it is important to know how much energy the power generators will need to produce in the future. This is because power generators take a significant amount of time to ramp up/down their power output [13]. If a significant amount of energy were to be available from wind turbines in the future, the power generators would need time to reduce their power output. It is therefore vital to have accurate wind power forecasting methods to schedule these generators to account for energy available from wind. This is currently an active area of research.

Wind generation forecasting algorithms can be sub divided into 2 groups: 1) Historical time series prediction methods. 2) Numerical Weather Prediction (NWP) models. NWP models incorporate weather data into the prediction of wind speed [14]. This research will focus on the former of the two groups using only historical time series wind data. AI methods are often referred to as grey box models. Grey box models combine mathematical based approaches with data to form the forecast model. Neural networks are typical examples of grey box models. There are many examples in the literature of applications of neural networks to wind energy forecasting. One such example is applying neural networks to predict wind power using local wind speed data [15]. Another application of neural networks to wind forecasting is implementing a multi neural network approach to first predict wind speed and to then map the wind speed to the power generated [16]. A comprehensive overview of the research surrounding wind power forecasting is given by Foley et al. [12].

2.3. Atmospheric pollutant forecasting

One of the major factors behind the rapid change in climate in recent years is the industrialization of the modern world. This is expected to increase as developing countries become more industrialized [17]. Climate change is considered to be one of the greatest threats facing modern society [18]. Man made climate change resulting from the emission of carbon dioxide emissions is responsible for many of the negative environmental changes observed in recent years. One of the most well known negative effects of climate change is the bleaching of coral reefs [19]. Increasing sea temperatures is believed to be responsible for this coral reef decline which is a direct result of climate change [20]. This warming of the oceans in turn has resulted in a decline in sea ice [21]. Less sea ice then results in increased sea levels [22]. One reason why this is problematic is that many of the worlds most populated cities are situated on the coast and are therefore more susceptible to flooding. The emissions of carbon dioxide is evidently a serious problem. It is therefore vital to develop accurate forecasting models to estimate CO_2 levels in both the short term and long term to enable better planning and control CO_2 levels.

In the context of power generation, a practical example of where the emission of greenhouse gasses is considered in the power generation process is the Dynamic Economic Emission Dispatch problem [13]. This is a multi-objective problem where a range of potential power generator configurations must be produced that minimize both cost and emissions to varying degrees. This range of solutions would then aid the engineer responsible for unit commitment and scheduling. This problem acknowledges that the environmental cost must also be considered in addition to the financial cost of power generation.

In terms of forecasting carbon dioxide levels, in 2015 Ganesan et al. implemented a neural network to forecast the exhaust emissions from a diesel electric generator [23]. A long term forecast of CO_2 levels from 1950 to 2050 was conducted by Schmalensee et al. that predicts how high global CO_2 levels will reach midway through the century [24]. Other studies have focused on the prediction of CO_2 levels at a national level in Brazil, Russia, India, China, and South Africa [25]. There are also many examples in the literature where other atmospheric pollutants are predicted. Bai et al. utilized neural networks to predict the daily air pollutant concentrations in China [26]. Krzywanski et al. predict the level of sulfur dioxide from boilers using a neural network [27]. Biancofiore et al. also utilized neural networks to predict hourly ozone concentrations [28]. The problem of predicting the levels of harmful atmospheric pollutants is an important task due to the adverse affects that these pollutants have on both the environment and the overall health of the population.

3. Neural networks

Neural networks were first proposed in the 1960s. Juergen Schmidhuber provides a comprehensive overview of the history of neural networks [29]. Neural networks are computational models that are inspired by the network of neurons that biological brain is comprised of. The function of these networks is to read in an input signal and produce an output signal that corresponds to that input. This functionality is useful for a wide range of problems including: robotics, control, classification, regression and time series forecasting. Neural networks are comprised of layers of processing units, referred to as neurons. The first layer, referred to as the input layer, reads in a signal to the network. Next are a number of hidden layers of neurons that pass the signal from the input layer through the network through weighted connections until the signal is outputted through the output layer. Each hidden neuron receives input in the form of weighted signals from the previous layer. All neurons possesses an activation function, the most common activation function being the sigmoid or logistic function (Eq. (1)).

$$a_j = \frac{1}{1 + \exp - v_j} \tag{1}$$

This activation function is used to calculate the neurons output signal a_j. The parameter v_j refers to the neurons input signal. This is calculated as shown in Eq. (2).

$$v_j = \sum_{i=1}^{N} w_{i,j} a_i \tag{2}$$

where v_j is the input to a neuron in the j^{th} layer, layer i is the preceding layer to layer j which contains N neurons, each neuron in the previous layer i has output a_i and each of these output signals are weighted by the value $w_{i,j}$ as they are passed to each neuron in the current layer j.

In terms of applying neural networks to forecasting, the input signal corresponds to the current and historic values in the time series. The network's output corresponds to future predicted values. Parameter tuning revealed that for the Irish energy time series data used in this research, 2 inputs provided the best performance, i.e. the current data point at time t and the previous data point at $t-1$. The output corresponds to the prediction for time $t+1$.

This research implements a particular form of neural network known as a Recurrent Neural Network (RNN) (Fig. 1). In the

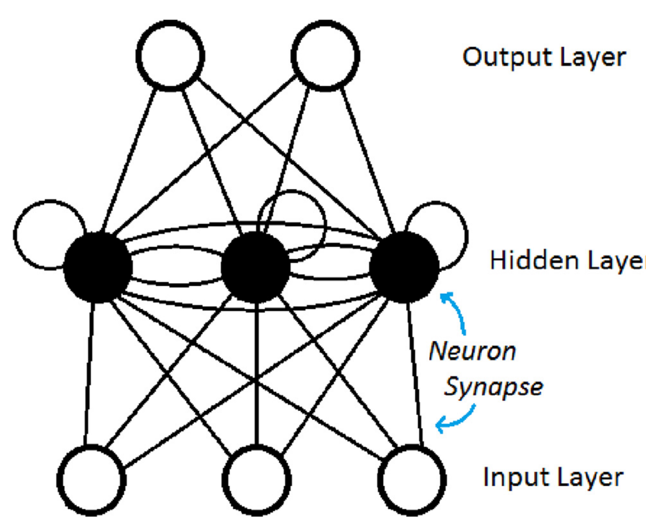

Fig. 1. Recurrent Neural Network [67]. This figure depicts a fully connected recurrent neural network. Neurons are connected by weighted connections (known as synapses). These connections pass signals between neurons. The recurrent connections can be seen in the hidden layer of neurons. These recurrent connections give the network memory from the previous forward pass.

standard feed forward neural network, each neuron receives input from the previous layer only. In recurrent networks however, the neurons in the hidden layer have connections joining neurons in the same layer. These are referred to as recurrent connections and give the system memory which is useful for problems with a temporal component such as time series forecasting.

A forward pass refers to when a signal is passed through the neural network and an output is produced. One of the main issues surrounding neural network research is how to set the weights so that a forward pass produces the correct output. The next section will discuss the backpropagation algorithm which is the most common method of training neural network weights.

There are many examples of neural networks being applied to predicting national power demands. In Iran, a multi-layered neural network was implemented to forecast monthly electrical energy consumption [30]. A neural network was applied to the task of forecasting long term energy forecasting in Greece [31]. The yearly energy demand was predicted in South Korea over a 27 year period using a neural network [32]. Another long term forecast study was conducted in Japan, where a neural network was trained on 20 years of data and then applied to forecast the subsequent 10 years [33]. In the USA, a neural network was utilized to forecast the energy consumption of the residential sector [34]. A hybrid neural network - linear model was implemented to forecast energy consumption in Taiwan [35]. There are also many examples of neural network forecasting in the literature with a focus on the environment. In 2018 Ye et al. explored the use of neural networks to predict CO_2 emissions from office buildings [36]. In 2017 Stamenkovic et al. utilized neural networks to predict nitrogen oxide emissions at a national level for 22 countries [37]. Another recent 2018 study has explored atmospheric dispersion prediction and estimated the source of hazardous gas using neural networks [38].

Many of the application of neural network to energy forecasting have been in the context of wind generation. In 2015, Osorio et al. implemented a neural network to forecast energy production of a wind farm in Portugal [39]. In 2016 Men et al. utilized ensemble neural networks to predict wind power and wind speed [15]. More recently, in 2017 Chang et al. proposed a radial basis function neural network-based model with an error feedback scheme to predict wind speed and power of a wind farm [40]. Each of these studies demonstrate the effectiveness of neural networks to predict the various factors associate with energy generation. The research presented in this paper builds upon each of these previous studies by applying neural networks to predict Irish power demand, wind energy generation and CO_2 levels.

In the context of research into Irish energy forecasting, there are no examples in the literature that utilize neural networks for time series forecasting. In 2004, Moehrlen explored the levels of uncertainty in wind energy forecasting using wind data from Ireland and Holland [41]. In 2005 Lang et al. implemented a Multi-Scheme Ensemble Prediction System to predict wind power generated from individual wind farms [42].

3.1. Backpropagation

This section will outline the Backpropagation algorithm which is the most common method of training neural networks. After a forward pass, the error E is calculated based on the difference between the target output t_i and the observed output a_i of unit k. The error E is calculated using Eq. (3).

$$E = \frac{1}{2}\sum_{k}(t_k - a_k)^2 \tag{3}$$

By taking the partial derivatives of the error w.r.t. each output and the partial derivatives of each output w.r.t. the network, the error delta (δ) for the final layer which is to be propagated can be calculated using Eq. (4).

$$\delta_k = (t_k - a_k)a_k(1 - a_k) \tag{4}$$

The weights between the final hidden layer and the output layer can then be updated using Eq. (5).

$$w_{j,k} = w_{j,k} + \alpha\delta_k a_j \tag{5}$$

where α is the learning rate. In order to propagate the error back through the hidden layers, the δ value for each hidden layer must be calculated. This can be done using the chain rule as the δ value for each neuron in layer j depends on the δ values of the neurons in layer $j+1$. The δ for each neuron in a hidden layer is calculated using Eq. (6).

$$\delta_j = \left[\sum_{j+1}\delta_{j+1}w_{j,j+1}\right]a_j(1 - a_j) \tag{6}$$

The weights between the hidden layer j and layer $j-1$ can then be updated using Eq. (7).

$$w_{j-1,j} = w_{j-1,j} + \alpha\delta_j a_{j-1} \tag{7}$$

Backpropagation is only suitable for problems that have target outputs, which is one of the limitations of the algorithm. It is however perfectly suited to time series forecasting and will be implemented as a benchmark to compare evolutionary neural networks to.

4. Swarm and evolutionary methods

The proposal of Genetic algorithms (GA) in the 1970s by John Holland was one of the first evolutionary algorithms proposed [43]. In the years since there have been a diverse range of proposed evolutionary algorithms. Some of the more popular and successful of these methods would include: Differential Evolution (DE), Particle Swarm Optimisation (PSO) and Covariance Matrix Adaptation Evolutionary Strategy (CMA-ES). The advantage of evolutionary strategies such as those listed above is that they are applicable to a wide range of optimisation problems including neural network training. These algorithms make no assumptions about the search space of the problem and are therefore applicable to problems with a range of challenging properties such as: noisy, multi-modal, non separable and very large problems. Traditional mathematical optimisation methods, such as gradient descent, struggle with these problems types. As a result, evolutionary methods are a popular choice of optimisation algorithm. Based on the extensive literature review conducted, there are no examples in the literature of CMA-ES being applied to the problem of forecasting in the energy sector. This is the primary contribution of this research. It is also apparent that the existing studies do not give a comprehensive experimental review of the different methods of evolving neural networks in the energy sector. The distinct contribution that this research makes is to compare and contrast the leading evolutionary algorithms on multiple energy forecasting problems. This research also makes the contribution of applying evolutionary neural networks to forecasting Ireland's energy needs.

4.1. Covariance matrix adaptation evolutionary strategy

One of the most studied optimisation algorithms in recent years is the Covariance Matrix Adaptation Evolutionary Strategy (CMA-

ES). CMA-ES was first invented in 1996 by Hansen and Ostermeier [44]. The algorithm was proposed as a method to solve non-convex, non linear and complex global optimisation problems. CMA-ES is classed as an evolutionary algorithm. These are algorithms that incorporate some of the various operators used in the evolutionary process that occurs in nature, e.g. crossover, mutation and selection. The CMA-ES algorithm begins by randomly sampling a number of solutions to the optimisation problem. For the task of optimising a neural network for forecasting problems, these solutions correspond to sets of network weights. These solutions are then ranked in accordance to their fitness. The mean solution (m), covariance matrix (C) and step size (σ) are then updated by increasing the chances of sampling solutions with better fitness scores. The pseudocode in Algorithm 1 outlines how this process is carried out. For a problem where the number of dimensions $= n$, the covariance matrix is an $n \times n$ matrix that determines the variance around the mean. For neural network optimisation, $n =$ the number of weights in the network to be optimised. In Algorithm 1, p_σ is the path for σ, p_c is the path for C, ω are the recombination weights such that $\omega_1 \geq \omega_2 \geq \ldots \geq \omega_\mu > 0$ sum to 1, y_ω is the move of the population mean, $\lambda = 10$ is the population size, μ is the number of samples selected for the update, I is the identity matrix and $N(m, \sigma^2 C)$ is the multivariate normal distribution. Parameter tuning found that $\sigma_0 = 0.1$ and $\mu = 0.5\lambda = 5$ gave the best performance. Standard values for other constants were implemented: $c_c = 4/n$, $c_\sigma = 4/n$, $c_1 = 2/n^2$, $c_\mu = \mu_\omega/n^2$, $d \approx 1$ and $\alpha = 1.5$.

CMA-ES has had a great deal of success in its application to many real world problems since its first proposal. CMA-ES has been applied to optimising building placement for solar energy [45]. A more high dimensional task that CMA-ES has proved to be successful at is the problem of laser pulse shaping [46]. The problem of antenna design has also benefited from the application of CMA-ES [47]. It has also been shown that a neural network trained with CMA-ES is an effective approach for reinforcement learning problems [48]. In 2013, Corne et al. hint at using CMA-ES for short term wind forecasting as future work [49]. The research presented in this paper pursues this suggested avenue of research. Based on the extensive literature review conducted, there are no examples of CMA-ES being applied to neural networks to forecast wind generation, energy demand or CO_2 levels.

Algorithm 1
Covariance Matrix Adaptation - Evolutionary Strategy Algorithm.

Initialize: Sample λ random solutions
Initialize: $C_0 = I$, $p_c = 0$ & $p_\lambda = 0$
while *Evaluation e <Emax* **do**
  for *Solution i = 1* ***to*** *λ* **do**
    $\vec{x_i} = m + \sigma y_i \sim N(m, \sigma^2 C)$
    $f_i = fitness(x_i)$
  end
  Sort solutions **x** according to fitness
  $m \leftarrow m + \sigma \sum_{i=1}^{\lambda} \omega_{p(i)} y_i =: m + \sigma y_\omega$
  $p_\sigma \leftarrow (1 - c_\sigma) p_\sigma + \sqrt{1 - (1 - c_\sigma)^2}\, \sqrt{\mu_\omega} C^{-0.5} y_\omega$
  $\sigma \leftarrow \sigma \times exp(\frac{c_\sigma}{d_\sigma}(\frac{\|p_\sigma\|}{E\|N(0,1)\|} - 1))$
  $p_c \leftarrow (1 - c_c) p_c \mathbb{1}_{[0, \alpha\sqrt{n}]}\{\|p_\sigma\|^2\} \sqrt{1 - (1 - c_c)^2}\, \sqrt{\mu_\omega} y_\omega$
  $C \leftarrow (1 - c_1 - c_\mu) C + c_\mu \sum_{i=1}^{\lambda} \omega_{p(i)} y_i y_i^T + c_1 p_c p_c^T$
end
Return best solution

4.2. Particle swarm optimisation

The next algorithm that will be applied to train the neural network is Particle Swarm Optimisation (PSO). The PSO algorithm dates back to 1995 when it was first proposed by James Kennedy [50]. The algorithm consists of a number $N = 50$ of particles. Each particle has a position $\vec{x_t}$ and velocity $\vec{v_t}$. A particle's position corresponds to its position in the search space, i.e. a candidate solution to the optimisation problem. For the task of neural network training, a position corresponds to a set of network weights. The velocity of the particle describes how the particle will change its position between iterations of the algorithm. For minimization problems, positions with a lower fitness are considered to be better. The fitness of a particle's position is determined by an objective function. For the problem of training a neural network for forecasting, this objective function corresponds to the Mean Squared Error. At each iteration, each particle will update its velocity based on its personal best position $\overrightarrow{pb}$ and the best overall position $\overrightarrow{gb}$. This process is carried out until a stopping criteria is met such as a threshold fitness being achieved or a predetermined number of evaluations have been carried out. Eq. (8) describes how the particles update their position and velocity.

$$\overrightarrow{v_{t+1}} = \chi(\vec{v_t} + r_1 c_1 \left(\overrightarrow{pb_t} - \vec{x_t}\right) + r_2 c_2 \left(\overrightarrow{gb_t} - \vec{x_t}\right), \tag{8a}$$

$$\overrightarrow{x_{t+1}} = \vec{x_t} + \vec{v_t} \tag{8b}$$

where c_1 and $c_2 = 2.05$ are acceleration coefficients, r_1 and r_2 are random numbers between 0 and 1 and finally $\chi \approx 0.72984$ is the constriction factor. The constriction factor guarantees that as the particles move around the problem space evaluating candidate solutions, they will eventually converge. The operation of the PSO algorithm is described in Algorithm 2.

Algorithm 2
Particle Swarm Optimisation Algorithm.

Initialize N particles with random $\vec{v_0}$ and $\vec{x_0}$
while *Evaluation e <Emax* **do**
  for *Particle = 1* ***to*** *N* **do**
    Update personal best position $\vec{pb_t}$
    Update neighbourhood best position $\vec{gb_t}$
    Evaluate particle's current position $\vec{x_t}$
    $\vec{v_{t+1}} = \chi(\vec{v_t} + r_1 c_1(\vec{pb_t} - \vec{x_t}) + r_2 c_2(\vec{gb_t} - \vec{x_t}))$
    $\vec{x_{t+1}} = \vec{x_t} + \vec{v_t}$
  end
end
Return best solution

PSO has been successfully applied to many real world problem domains. A PSO trained neural network has shown to be a successful strategy for the task of watershed management [51]. It has also been shown that PSO is very effective at the task of power generator scheduling [13]. In 2009, Welch et al. implemented PSO to train neural networks for wind speed forecasting with great success [52]. In 2011, Catalao et al. applied PSO in conjunction with fuzzy logic and a wavelet transform to predict wind power generation in Portugal [53]. PSO has also been applied to predicting power demand in Iran in 2014 by Bahrami et al. [54]. This study did not however implement neural networks as the predictive model. Similarly in 2014, Selakov et al. combined PSO with Support Vector Machines (SVM) to predict power demand for the city of Burbank in the USA [55]. In 2009, PSO was implemented with neural networks for short term load forecasting for an area in New York [56]. The authors concluded that PSO was in fact an effective neural network training algorithm. There has been much less research in applying PSO to forecasting CO_2 levels. In 2016, Ozceylan applied PSO to CO_2 emissions forecasting in Turkey [57]. This study did not investigate combining PSO with neural networks, which is element of this research. The research presented in this paper builds on these previous studies by applying PSO to neural networks and applying the resulting forecasting algorithm to wind, energy and CO_2 forecasting in Ireland. This research will also compare PSO to other popular evolutionary algorithms that were not evaluated in many of these previous studies.

4.3. Differential evolution

The final evolutionary algorithm that will be implemented in this study is Differential Evolution (DE). Like CMA-ES, DE uses evolutionary operators such as selection, crossover and mutation to optimise complex global optimisation problems. The algorithm was first proposed in 1997 by Storn and Price [58]. Like all evolutionary algorithms, DE does not rely on problem gradients or need any prior information about the problem at hand. This makes it a highly robust and widely applicable optimisation algorithm that can be applied to problems that traditional mathematical optimisation algorithms would either take too much computational time to solve or would converge onto a local optimum. DE is also a more straightforward optimisation algorithm to implement compared to CMA-ES. The ease of implementation and effectiveness of DE makes it a popular optimisation algorithm and suitable for the task of optimising network weights for forecasting problems.

The DE algorithm begins by creating a number of agents with random position in the problem space. Like the particles in PSO, the agents in DE each have a position $\vec{x}$ that represents a potential solution. The solution here represents a set of network weights. At each iteration of the algorithm, the current agent's position is combined with the positions of three other distinct agents' positions to produce a new position $\vec{y_i}$. This is outlined in the pseudo-code in Algorithm 3. If the new position $\vec{y_i}$ of the agent is better than its previous position $\vec{x_i}$, the agent moves to the new position. The quality of the position is judged based on its fitness which is calculated using the objective function. In this research, the fitness function is the MSE of the neural network when evaluated on the training data. This is repeated for each of the agents until a predetermined number of problem evaluations has been conducted or until some fitness threshold has been reached. In Algorithm 3 the parameter $CR = 0.7$ is the crossover probability, $F = 0.5$ is the differential weight and N is the number of agents ($N =$ the number of network weights).

Within the context of energy forecasting, Meyyappan et al. applied DE to neural networks for wind power prediction for economic dispatch in 2015 [59]. In 2016, Yang et al. applied DE to neural networks to predict the short term demand in New South Wales, Australia [60]. These previous studies are built upon in this paper by applying DE to forecast power demand, wind generation and CO_2 for Ireland. Based on the extensive literature review conducted, there are no examples in the literature of DE being applied to predict CO_2 levels.

5. Experimental methods

This section will outline the various experiments conducted and the implementation details of each algorithm. The experiments described in this research were implemented in Java.

5.1. Forecast data

Each forecasting method will be evaluated on data sets relating to Ireland's: 1) Energy demand. 2) Wind power generation. 3) CO_2 intensity levels. In each of these three forecasting problems, each forecasting algorithm will be trained on a month of data from the 16^{th} of August 2017 until the 14^{th} of September 2017. The performance of each model will then be tested on completely separate month of time series data sets from the 18^{th} of September 2017 until the 17^{th} of October 2017. Each of these data sets consist of time series data in increments of 15 min, with a total of 2878 data points. This data was sourced from Eirgrid [61].

Algorithm 3
Differential Evolution Algorithm.

```
Initialize N agents with random positions x0
while Evaluation e <Emax do
  for Agent = 1 to N do
    Select 3 other agents A,B and C
    Select random dimension index R
    for dimension i = 1 to D do
      generate random number r ∈ [0,1]
      if r < CR Or i = R then
          new position y_i = a_i + F × (b_i − c_i)
      else
          y_i = x_i
      end if
    end
    if fitness(y) < fitness(x) then
       replace x with y
    end if
  end
end
Return best solution
```

5.2. Network parameter selection

A recurrent neural network was implemented in this research for time series prediction. Recurrent neural networks were selected over feed forward networks as the recurrent connections give the network a memory of previous forward passes. This is beneficial for time series problems as there is a strong temporal element in time series data sets, i.e. a correlation between the current state time step and previous time step. Each network is fully connected and has six hidden neurons. The networks have 2 inputs corresponding to the current (time t) and previous (time t-1) time step values. This was implemented for all data sets. Parameter sweeps revealed that 6 was the optimum number of hidden neurons. More than 6 hidden neurons did not lead to any increase in performance due to the increased number of weights that must be trained as the network increased in size. Parameter sweeps also established that more than two network inputs did not lead to any increase in performance. The networks have one output corresponding to the network's prediction at time step t + 1.

5.3. Network training

Each network with 6 neurons has a corresponding set of 54 weights. All neural network training algorithms optimised the set of weights for 10E6 evaluations of the training data. The fitness of the network was determined by the Mean Squared Error (MSE) of networks prediction accuracy on the training data. After a network is trained for 10E6 evaluations on the training data, it is then evaluated on the test data. This process is repeated over 10 runs to ensure statistically significant results.

5.4. Comparative forecasting methods

In total 7 algorithms will be evaluated on each of the 3 forecasting problems. These are:

1. Covariance Matrix Adaptation Evolutionary Strategy (CMA-ES).
2. Particle Swarm Optimisation (PSO).
3. Differential Evolution (DE).
4. Back Propagation (BP).
5. Moving Average (MA).
6. Random walk forecasting (RWF).
7. Linear Regression (LR).

Implementing this diverse set of 7 algorithms gives a comprehensive comparison of forecasting algorithms for energy time series problems. Forecasting methods 1–4 all involve training a neural network to output the predicted values. The implementation details of each of these algorithms has been outlined in detail in Sections 3 and 4.

The final three algorithms: Moving Average, Random Forest Walk and Linear Regression will be implemented to give further context to the neural network based forecasting methods. The moving average method is a commonly used forecasting approach. This method consists of predicting a future value as an average of n previous values. The forecast value at time t+1, V_{t+1} is calculated using Eq. (9).

$$V_{t+1} = \frac{\sum_{i=0}^{n-1} V_{t-n}}{n} \tag{9}$$

Random walk forecasting is the next forecasting method that will be implemented. This approach simply consists of predicting the next future value V_{t+1} as equal to the current observed value V_t.

Linear regression (LR) is a well known machine learning algorithm that is routinely used for long term electricity consumption forecasting [62]. Linear regression consists of constructing a linear model whereby the parameters of the model are estimated using the data [63]. Eq. (10) describes the linear model that LR implements to forecast the future value V_{t+1} at time $t+1$. The parameters C are constants that are adjusted by the LR algorithm so that the model produces the lowest forecasting error.

$$V_{t+1} = C_1 V_{t-1} + C_2 V_t + C_3 \tag{10}$$

5.5. Forecast accuracy evaluation

Four standard metrics will be used to evaluate the performance of each forecasting algorithm. The first is the Mean Absolute Error (MAE). This is an easy to understand metric which has the added benefit of having the same units as the value being predicted. The MAE is calculated by Eq. (11).

$$MAE = \frac{\sum_{i=1}^{n} |y_i - x_i|}{n} \tag{11}$$

The next metric is the Mean Squared Error (MSE). This metric does not have the same units as the value being predicted. The MSE is more sensitive to outliers in the data and is calculated using Eq. (12).

$$MSE = \frac{\sum_{i=1}^{n} |y_i - x_i|^2}{n} \tag{12}$$

The Root Mean Squared Error (RMSE) is the next metric that will be evaluated. Like the MSE, it is more sensitive to outliers than the MAE, however it is more readable than the MSE. The RMSE is calculated using Eq. (13).

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n} |y_i - x_i|^2}{n}} \tag{13}$$

The final metric that will be evaluated is the Mean Absolute Percentage Error (MAPE). This metric gives the forecasting error as a percentage of the actual value being estimated, this means that the MAPE can be used to compare forecasting approaches across problems with different units. The MAPE is calculated using Eq. (14).

$$MAPE = \frac{100}{n} \sum_{i=1}^{n} \frac{|y_i - x_i|}{|y_i|} \tag{14}$$

where n is the number of forecast values, y_i is the actual value and x_i is the forecast value.

5.6. Experiments conducted

There will be three experiments conducted in this research paper. The first experiment will involve evaluating the performance of each forecasting algorithm on each set of training data. All algorithms will be judged based on their rate of convergence, prediction accuracy, performance consistency and ease of implementation.

The second experiment will evaluate the performance of each trained network on a month of previously unseen data for each of the three forecasting problems. The purpose of this is to establish the generality of each forecasting method and ensure they are capable of providing good performance outside of the training data.

The third and final experiment will evaluate how far into the future the evolved network can predict. In terms of planning, it would be advantageous to have accurate predictions farther into the future. This would aid the scheduling and operation of the power generation process.

6. Results

This section presents the results of each of the experiments outlined above followed by a discussion in order to gain insight and highlight their significance. The two tailed t-test with a significance level of $\alpha = 0.05$ was conducted to determine significant performance differences when comparing algorithms. All values were rounded to 4 significant figures.

6.1. Convergence

The convergence of CMA-ES, PSO and DE can be seen in Fig. 2(a)–(c) for the wind power generation, power demand and CO_2 emissions respectively. In each of these graphs, it can be seen that CMA-ES and PSO converge the fastest, however PSO stops improving after approximately 10E5 evaluations for each problem. CMA-ES continuously improves for the entire training period. These improvements are much more incremental after approximately 10E5 evaluations. DE is the slowest converging algorithm of the evolutionary approaches. DE offers much more gradual improvements in the accuracy of the network and eventually converges to a better solution than PSO on the power demand prediction problem (Fig. 2 (b)). Of these three evolutionary algorithms, CMA-ES and PSO provide the fastest convergence. Overall CMA-ES provides the best performance in terms of rate of convergence and final network accuracy on the training data.

In terms of complexity, DE and PSO are more straightforward to implement than CMA-ES. Although these algorithms are more simple in their design, Fig. 2 illustrates that the added complexity of CMA-ES does result in a more accurate neural network for forecasting. It can be seen that PSO does provide rapid convergence. Therefore if limited time or computational resources are available, PSO could be a viable alternative. Although DE is slower to converge, it is apparent from the convergence graphs that it has not yet converged on an optimum solution. Given more computational time, DE could perhaps converge on a network with an accuracy comparable with CMA-ES.

6.2. Accuracy

The accuracy of each evolutionary method and other benchmark algorithms for both training and test data is presented in Table 1 for the wind power generation, power demand and CO_2 emissions. In this table, the average and standard deviation of the MAE, MSE, RMSE and MAPE are displayed for both the training and test sets. The algorithm with the lowest error is highlighted in bold in Table 1. It is evident from this table that a network trained with CMA-ES performs significantly better than other methods on the training data. CMA-ES provides the highest prediction accuracy on all three sets of training data. On the test data sets, CMA-ES performs best on two of the three forecasting problems (wind power generation and power demand). CMA-ES performs third best on the CO_2 intensity level data set, where it is outperformed by LR and RW. Of the neural network based approaches, CMA-ES performs best on all data sets, for training and testing. In terms of performance consistency, BP had the smallest standard deviation followed by CMA-ES, PSO then DE. This is the same across all forecasting problems. Of the non neural network based approaches, LR had the highest forecasting accuracy on the wind power generation and CO_2 intensity prediction problems, followed by RW then MA. On the power demand prediction problem, LR performed the worst. The reasons for this will be discussed in detail in Section 7.

One observation from Table 1 worth noting is the difference between the MAPE and MAE, MSE and RMSE on the wind generation prediction problem. CMA-ES performs significantly better than all other approaches when compared using the MAE, MSE or RMSE. CMA-ES performs slightly worse than LR however when compared using the MAPE. This points to a common criticism of the MAPE metric. The MAPE metric gives higher significance to errors when the target value is near 0 than values that are not. For example a forecast error of 10 MW would correspond to a percentage error of 0.5% if the value being forecast $y_i = 2000$ MW. However a forecast error of 10 MW would correspond to a percentage error of 100% if the value being forecast $y_i = 10$ MW. The wind power prediction problem contains values that are near zero. This means that MAPE gives a higher significance to these lower values than others. Since CMA-ES has a higher MAPE than LR on this problem, it indicates that LR is slightly better at predicting these lower values. However if it is assumed that all time series values have equal significance, which it is believed to be the case, then the

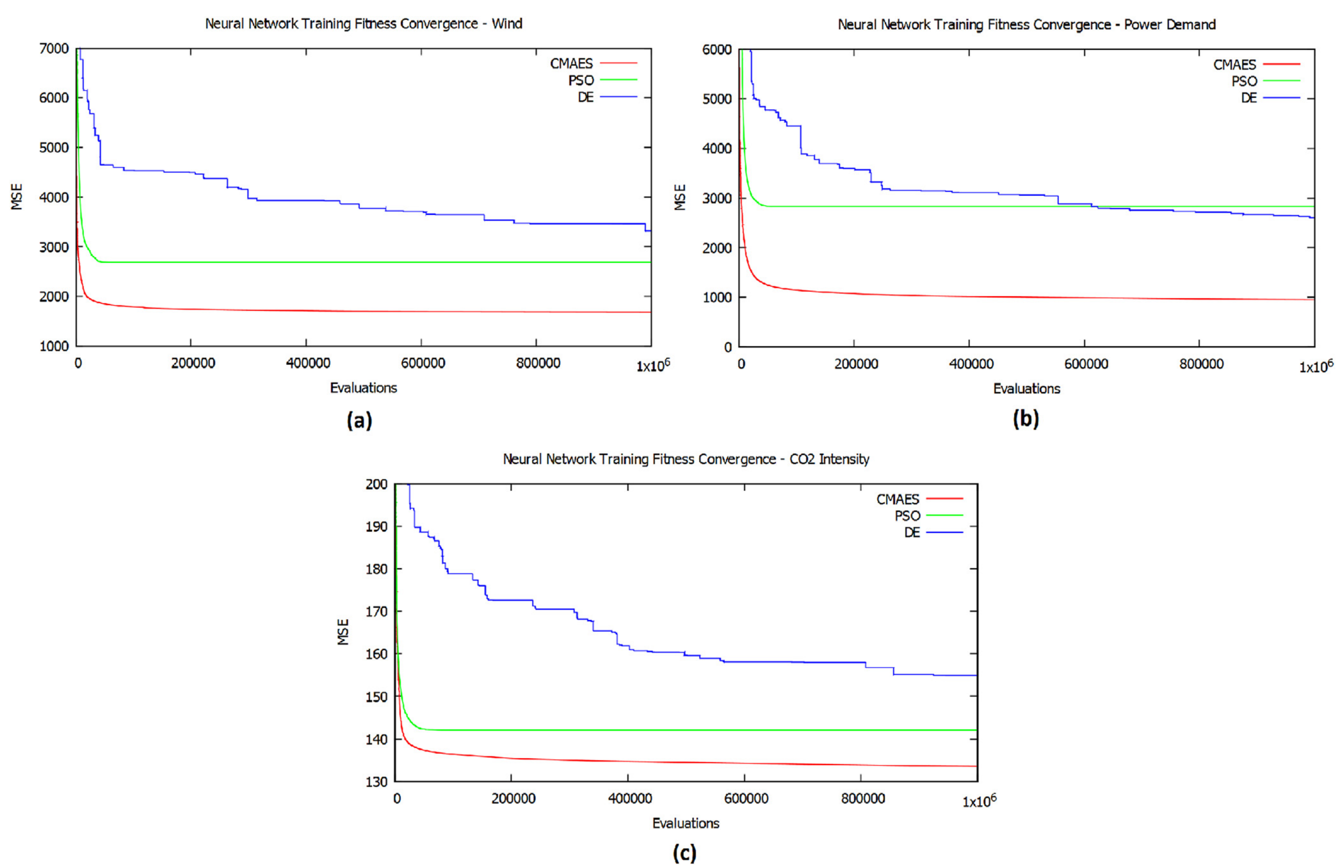

Fig. 2. Convergence of Mean Squared Error for Each Neural Network Training Algorithm. This figure illustrates the convergence of each evolutionary neural network training algorithm as each network learns to predict the training data. Graphs a, b and c depict the wind power, power demand and carbon dioxide levels respectively.

Table 1
Forecast accuracy.

Algorithm	Wind Power Generated							
	Training				Testing			
	MAE (Std Dev)	MSE (Std Dev)	RMSE (Std Dev)	MAPE (Std Dev)	MAE (Std Dev)	MSE (Std Dev)	RMSE (Std Dev)	MAPE (Std Dev)
CMA-ES	**29.70 (0.518)**	**1684 (42.65)**	**41.04 (0.517)**	5.021 (0.657)	**32.88 (0.886)**	**2105 (79.95)**	**45.87 (0.859)**	6.341 (1.540)
PSO	38.93 (1.613)	2689 (197.4)	51.82 (1.898)	11.08 (1.614)	48.12 (2.670)	4060 (467.1)	63.61 (3.623)	18.20 (3.176)
DE	42.71 (3.427)	3320 (508.2)	57.44 (4.492)	11.35 (4.177)	50.07 (4.942)	4545 (1003)	67.00 (7.497)	17.97 (8.095)
BP	33.67 (0.004)	2053 (0.366)	45.31 (0.004)	8.313 (0.004)	38.655 (0.005)	2627 (0.545)	51.25 (0.005)	12.50 (0.007)
MA	39.37 (0.000)	2943 (0.000)	54.25 (0.000)	5.572 (0.000)	48.01 (0.000)	4482 (0.000)	66.95 (0.000)	6.069 (0.000)
RW	31.98 (0.000)	1954 (0.000)	44.20 (0.000)	**4.524 (0.000)**	36.81 (0.000)	2678 (0.000)	51.75 (0.000)	4.645 (0.000)
LR	30.83 (0.000)	1821 (0.000)	42.67 (0.000)	**4.524 (0.000)**	33.85 (0.000)	2268 (0.000)	47.62 (0.000)	**4.516 (0.000)**
Algorithm	Power Demand							
	Training				Testing			
	MAE (Std Dev)	MSE (Std Dev)	RMSE (Std Dev)	MAPE (Std Dev)	MAE (Std Dev)	MSE (Std Dev)	RMSE (Std Dev)	MAPE (Std Dev)
CMA-ES	**22.50 (0.190)**	**951.8 (25.10)**	**30.85 (0.409)**	**0.581 (0.005)**	**24.47 (0.370)**	**1133 (46.71)**	**33.66 (0.692)**	**0.610 (0.009)**
PSO	39.42 (1.856)	2832 (231.6)	53.17 (2.155)	1.064 (0.065)	42.88 (1.700)	3429 (326.1)	58.50 (2.757)	1.091 (0.049)
DE	37.88 (3.139)	2606 (424.2)	50.87 (4.270)	1.022 (0.094)	43.77 (4.573)	3713 (783.6)	60.60 (6.362)	1.102 (0.111)
BP	26.65 (0.014)	1354 (71.72)	36.78 (0.945)	0.831 (0.004)	29.97 (0.062)	1684 (6.459)	41.04 (0.079)	12.50 (0.007)
MA	72.42 (0.000)	9758 (0.000)	98.78 (0.000)	1.938 (0.000)	77.35 (0.000)	11371 (0.00)	106.6 (0.000)	1.998 (0.000)
RW	49.95 (0.000)	4629 (0.000)	68.04 (0.000)	1.334 (0.000)	53.19 (0.000)	5371 (0.000)	73.29 (0.000)	1.373 (0.000)
LR	97.41 (0.000)	17067 (0.00)	130.6 (0.000)	2.616 (0.000)	104.3 (0.000)	19886 (0.00)	141.0 (0.000)	2.701 (0.000)
Algorithm	CO_2 Intensity Level							
	Training				Testing			
	MAE (Std Dev)	MSE (Std Dev)	RMSE (Std Dev)	MAPE (Std Dev)	MAE (Std Dev)	MSE (Std Dev)	RMSE (Std Dev)	MAPE (Std Dev)
CMA-ES	**8.879 (0.052)**	**133.6 (1.472)**	**11.56 (0.064)**	**2.056 (0.012)**	8.200 (0.391)	120.5 (17.33)	10.98 (0.743)	2.258 (0.152)
PSO	9.216 (0.101)	142.1 (3.653)	11.92 (0.152)	2.130 (0.018)	9.035 (0.157)	143.2 (6.098)	11.96 (0.251)	2.558 (0.058)
DE	9.614 (0.266)	155.0 (7.605)	12.44 (0.310)	2.231 (0.064)	9.669 (0.872)	165.5 (36.57)	12.79 (1.360)	2.747 (0.319)
BP	9.178 (0.000)	140.4 (0.024)	11.85 (0.001)	2.124 (0.005)	8.569 (0.021)	125.1 (0.614)	11.18 (0.021)	2.386 (0.008)
MA	9.290 (0.000)	145.1 (0.000)	12.04 (0.000)	2.145 (0.000)	8.316 (0.000)	120.1 (0.000)	10.96 (0.000)	2.236 (0.000)
RW	9.304 (0.000)	148.7 (0.000)	12.20 (0.000)	2.162 (0.000)	8.011 (0.000)	111.2 (0.000)	10.55 (0.000)	2.168 (0.000)
LR	8.993 (0.000)	137.0 (0.000)	11.70 (0.000)	2.082 (0.000)	**7.924 (0.000)**	**109.1 (0.000)**	**10.45 (0.000)**	**2.140 (0.000)**

MAE, MSE and RMSE are better indicators of forecast accuracy. These three metrics reveal that CMA-ES has a higher accuracy than LR. MAPE does indicate that the accuracy of CMA-ES might slightly decrease when predicting lower values.

When inspecting the differences between the accuracy of each forecast model on the training and test data, it is evident that each model generalizes well to unseen data. As Table 1 shows, the accuracy of each model is only slightly worse on the test data than on the training data for the wind and power demand forecast problems. The accuracy of most models actually improves on the CO_2 forecasting test data. A possible reason for this is that lower CO_2 values are observed overall on the month of test data than the month of training data. It appears to be the case that the forecast models are better at forecasting these lower values for the CO_2 data set. The PSO and DE trained networks seem to be an exception to this however as they have higher MSE for the CO_2 test data than for the training data.

Fig. 3 illustrates the spread of each forecasting algorithm. The most salient observation from these graphs is that each of the evolutionary neural networks has much higher variance in its performance. CMA-ES has the smallest spread, followed by PSO then DE. This is due to the stochastic nature of these algorithms. The performance of DE on the CO_2 forecasting problem is one of the more noticeable features in Fig. 3 (c). DE has a large amount of variation in its accuracy on the test data for each run. The worst run is significantly worse than all other approaches. Its best MAE however is comparable with CMA-ES, LR and RW which have the highest accuracy. CMA-ES is also very inconsistent with its performance on the CO_2 test data. This is also the only problem where CMA-ES does not have the best average MAE, MSE or RMSE. The reason for this could be the previously mentioned lower test CO_2 values. This will be discussed further in Section 7.

6.3. Predictions

The forecast predictions of each evolutionary algorithm are plotted along with the actual values for the training and test sets in Figs. 4–6 of the wind, power demand and CO_2 data sets respectively. As these graphs illustrate, each forecasting algorithm is capable of forecasting future values with a high degree of accuracy. Differences between the forecast values and the actual values can only be observed at the peaks of the time series data. This highlights that the error of the forecasting methods is largest when there is a rapid change in the time series data. This can be observed in all of the forecasting problems in Figs. 4–6.

There is also an interesting distinction between the forecast values in the training set and the test set. As Fig. 6 illustrates, there is a much more noticeable difference between the forecast values and the actual values in the test data (Graph (b)) than in the training data (Graph (a)) for the PSO and DE trained networks. This is particularly evident for the lower CO_2 intensity values. As previously mentioned, DE and PSO had the highest error on this data set, which highlight that they do not generalize to unseen CO_2 data as well as the other models. The suspected reason for this is that there are significantly lower CO_2 values in the test set than in the training set. This seems to favor other forecast models including the CMA-ES and BP trained networks, however the PSO and DE trained networks produce less accurate forecasts for lower CO_2 levels. As could be seen in the convergence graphs in Fig. 2, PSO and DE converge to produce a network with a much lower accuracy than CMA-ES. It is thought that these higher CO_2 forecast errors in the test set is a manifestation of this.

The generalization of each algorithm is discussed in more detail in Section 7. When inspecting the differences between the forecast and actual values in the CO_2 test data (Fig. 6 (b)), it is observed that

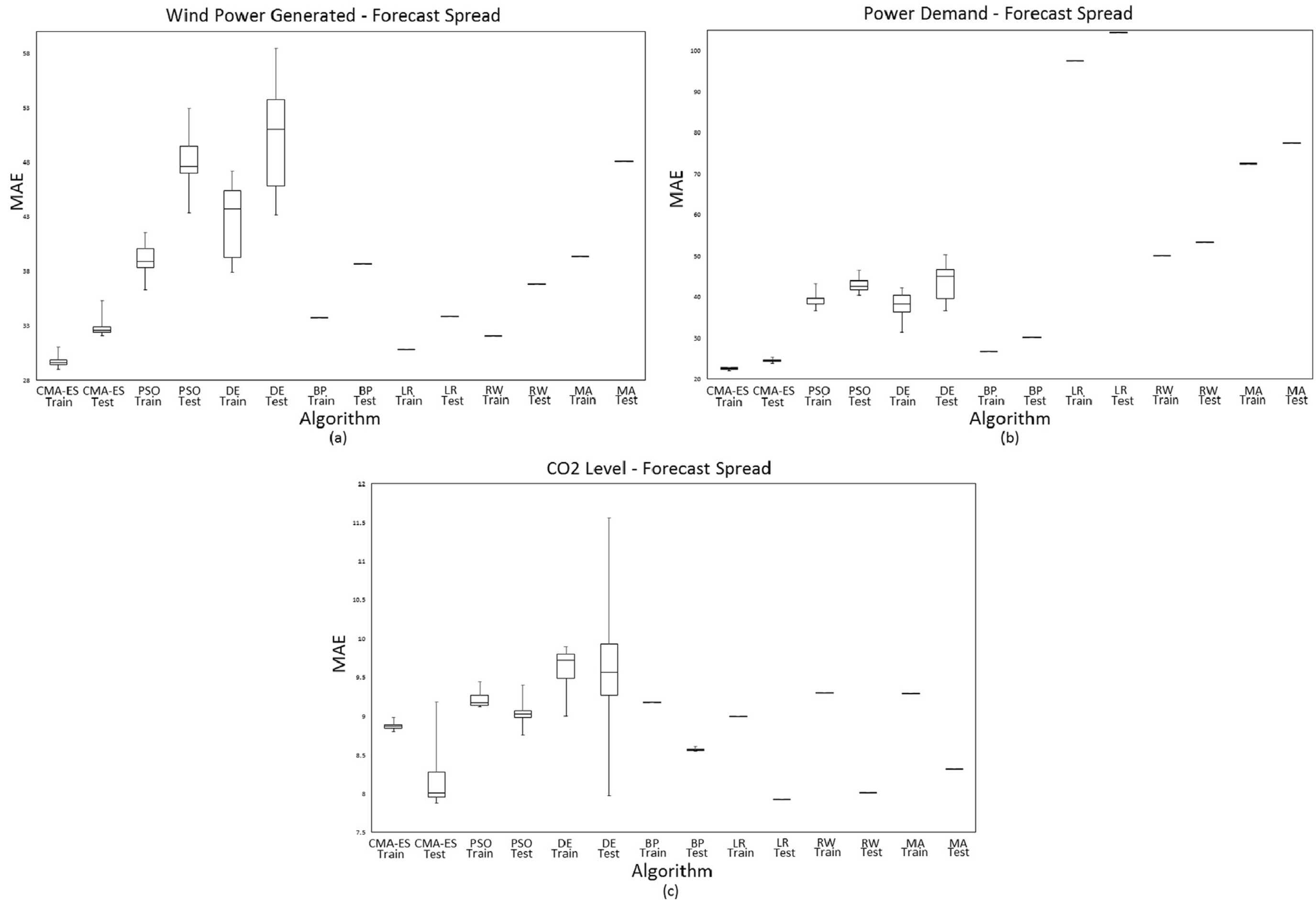

Fig. 3. Forecast Accuracy Spread of Each Algorithm. This figure illustrates the spread in accuracy of each forecasting algorithm. Graphs a, b and c depict the wind power, power demand and carbon dioxide levels respectively.

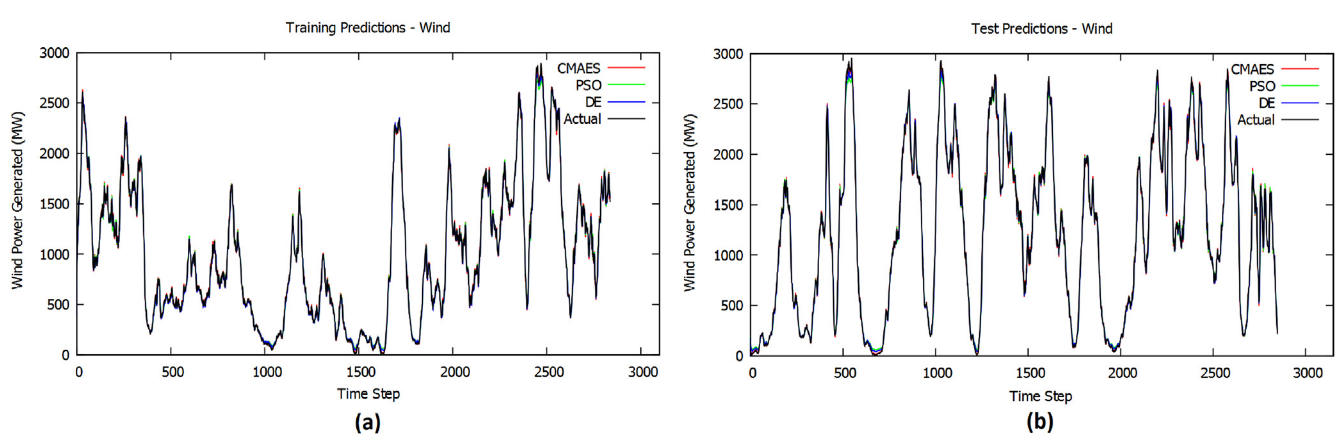

Fig. 4. Wind Power Generation Predictions for Training and Test Data. This figure illustrates the predicted wind power generated at each time step using the evolved neural networks. Graph (a) illustrates the predictions for the training data while graph (b) presents the test data predictions.

DE has the largest difference from the actual CO_2 values, followed by PSO then CMA-ES. This reflects the overall accuracy of each algorithm in Table 1. Unlike PSO and DE, CMA-ES has a higher overall accuracy in the test set than the training set for the CO_2 prediction problem. This can be seen in Fig. 6, where the predictions of CMA-ES is much closer to the actual values than PSO and DE.

The error at each time step for each forecasting problem can be seen in Figures A.8, A.9 and A.10 in Appendix A. These graphs highlight that the primary source of error for CMA-ES evolved neural network is when there are sudden extreme changes in the value being predicted. As expected the network has the highest accuracy when there is a shallow gradient in the time series data. The wind power forecasting error in Figure A.8 clearly illustrates this. At approximately time step 1500 of the training data and 2000 of the test data, the error on the lower plot is very low. These points correspond to locations on the upper time series plots of the wind

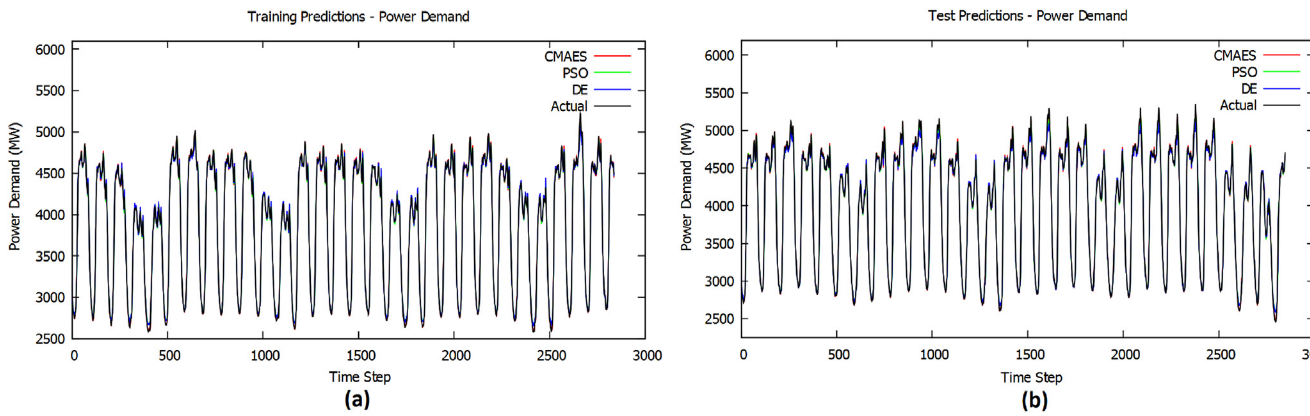

Fig. 5. Power Demand Predictions for Training and Test Data. This figure illustrates the predicted power demand at each time step using the evolved neural networks. Graph (a) illustrates the predictions for the training data while graph (b) presents the test data predictions.

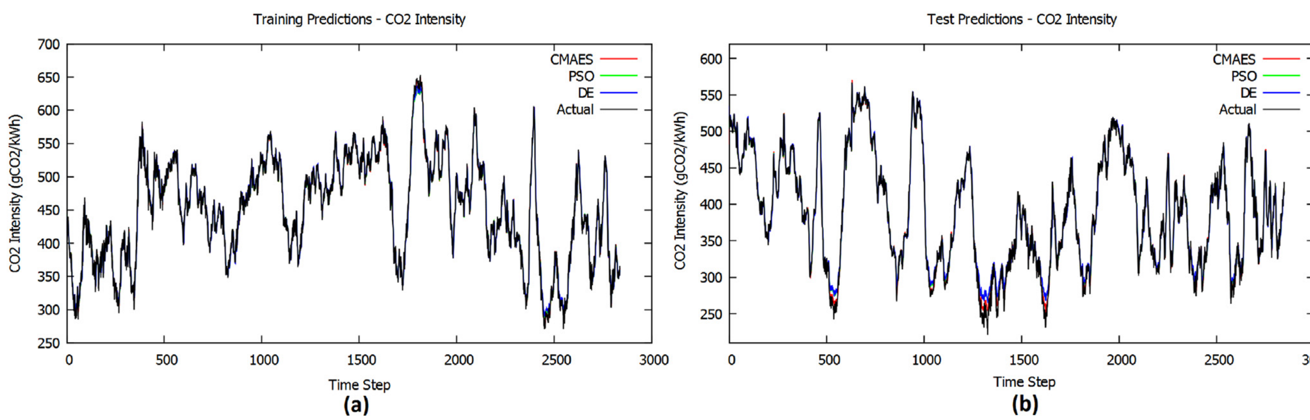

Fig. 6. Carbon Dioxide Intensity Predictions for Training and Test Data. This figure illustrates the predicted carbon dioxide level at each time step using the evolved neural networks. Graph (a) illustrates the predictions for the training data while graph (b) presents the test data predictions.

power where there is very little change in the power generated from wind. The error spikes in the lower plots correspond to locations in the time series data where there are sudden changes in wind power generation. There is a particularly large error at approximately time step 2700 in the training set. The wind power generated at this time changes from a steep decline to a steep incline. The stochastic nature of the power available from wind means that it is very difficult for the neural network forecasting model to anticipate and adapt to these sudden random changes. For this reason many forecast models include wind speed forecasts into the prediction of wind power generation. As previously stated however, this is outside of the scope of this research.

As Figure A.9 illustrates, there is much less stochasticity in the power demand profile. It is periodic in nature unlike the wind power generation and CO_2 level profiles. The error in the power demand predictions is also therefore regular. Every day at approximately 6:00 a.m., when the power demand changes from decreasing to increasing, there is a spike in the prediction error. Despite the fact that this is a regular occurrence every day, the accuracy of the neural network prediction is at its lowest at this point. The CMA-ES trained neural network does however have a higher accuracy than all other approaches evaluated for power demand forecasting. One possible way to increase the accuracy of the neural network at forecasting the power demand would be to incorporate the time of day as an input to the network. This would give the network a sense of the periodic nature of the problem and potentially improve performance.

Finally, Figure A.10 graphs the error at each time step for the CO_2 intensity prediction problem. Similar to the wind power generation graph, there is no periodicity in the CO_2 data. Similar observations can be made from this graph as from the wind power graph, the prediction errors correspond to points in time when there is the a rapid change in CO_2 intensity. There is a particularly high forecast error at time step 627 of the test data. As expected, this is due to a very rapid change in the CO_2 levels. From time step 622 to 626 the CO_2 intensity level was steadily decreasing from approximately 485 to 472 g CO_2/kWh, which the neural network was able to predict with a reasonable degree of accuracy. Within the 15 min between time point 626 and 627, the CO_2 intensity increases from 472 to 546 g CO_2/kWh. However the network had predicted that the CO_2 levels would decrease at a similar steady rate that it had observed for the previous 4 time steps. This results in the relatively large prediction error observed in test predictions.

6.4. Multi-step ahead prediction

The aim of the experiment conducted in this section was to evaluate how the forecast accuracy of the neural network changes as it attempts to predict farther into the future. Fig. 7 illustrates how the forecasting accuracy decreases as it predicts 1 time step into the future (15 min) versus 10 time steps (2.5 h). As this graph demonstrates, there is a predominantly linear increase in forecasting error (MAE) as the network trained with CMA-ES predicts farther into the future. This is observed for all data sets. The CO_2

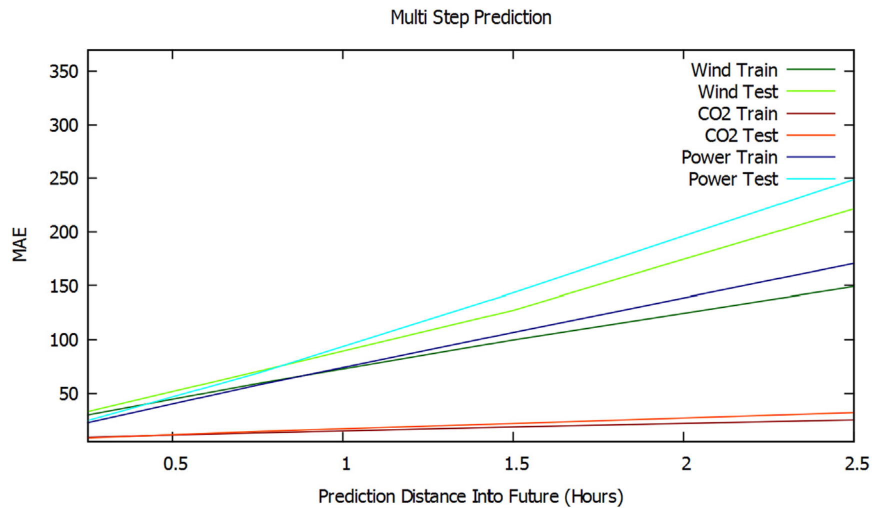

Fig. 7. Mean Absolute Error of Covariance Matrix Adaptation - Evolutionary Strategy trained Neural Network When Forecasting Further Into the Future. This figure illustrates how the forecasting accuracy of the neural network degrades as it predicts further into the future for each problem.

data set has different units to the wind and power demand data sets, which is why the increase appears to be much less than for the other two data sets. The prediction accuracy for the CO_2 data set increases from $MAE_{Train} = 8.879$ and $MAE_{Test} = 8.200$ for 1 step ahead prediction to $MAE_{Train} = 25.106$ and $MAE_{Test} = 31.964$ for 10 steps ahead prediction. The accuracy for the wind forecasting problem changes from $MAE_{Train} = 29.701$ and $MAE_{Test} = 32.819$ for 1 step ahead prediction to $MAE_{Train} = 149.176$ and $MAE_{Test} = 221.482$ for 10 steps ahead prediction. Finally the forecasting accuracy for predicting the power demand changes from $MAE_{Train} = 22.495$ and $MAE_{Test} = 24.466$ for 1 step ahead prediction to $MAE_{Train} = 170.584$ and $MAE_{Test} = 248.785$ for 10 steps ahead prediction. It is to be expected that the forecasting accuracy decreases as the algorithm attempts to predict farther into the future. Given that values that are to be predicted vary by 1000s of MWs for the power and wind forecasting problems, the 2.5 h ahead forecasting accuracy achieved by the CMA-ES is still considered relatively accurate.

Another observation that can be made from Fig. 7 is that difference between the forecasting accuracy on the training and test data becomes larger as the network predicts farther ahead. The forecasting accuracy difference is almost indistinguishable for 1 time step ahead but is significantly higher for the training data on all three problems when predicting 10 steps ahead. This indicates that the networks are able to more accurately forecast previously unseen data when predicting only 1 step ahead.

7. Discussion

The results presented in the previous section demonstrate that evolving neural networks using CMA-ES is an effective approach to developing accurate forecasting models. In terms of accuracy, convergence and robustness, CMA-ES outperforms all other approaches evaluated for each of the three energy forecasting problems. CMA-ES provides the highest forecasting accuracy for the training data in all three forecasting problems. When evaluated using previously unseen test data, CMA-ES performs best on two of the three forecasting problems. When contrasting the performance differences between the training and test data, it is insightful to look at the training and testing data sets. CMA-ES had the third highest MAE on the test data for the CO_2 intensity forecasting problem. It is thought that the reason CMA-ES was outperformed in this instance by RW and LR is that the test data set contained CO_2 levels that were significantly lower than any values in the training set. Although every model aside from DE and PSO trained networks produced a lower forecast MSE on the CO_2 test data, it appears that LR is slightly more robust to unseen data that is significantly different from data experienced during training than the neural network based approaches.

Fig. 6 illustrates the increased error when the PSO and DE trained neural networks predict these lower values in the test set. The training set didn't contain any values lower than 250 g CO_2/kWh, while the test set contained three separate instances where the CO_2 level drops below 250 g CO_2/kWh. The PSO and DE trained neural networks produced higher errors when predicting these values. This is a well known problem in machine learning research however. The performance of any function approximators will suffer if it is given new values that are significantly different from those previously experienced in training. A way to mitigate this issue would be to train the networks on a larger data set containing every possible values that the network could possibly face when implemented to forecast for new and unseen data. Such a data set is currently unavailable to the authors and is therefore differed as a topic for future research. CMA-ES did still perform third best on the test data for the CO_2 data but it is worth highlighting this issue. LR appears to be more robust to significantly different values for the CO_2 forecasting problem.

This issue was not observed in the test data for the wind power generation and power demand prediction problems. As is evident from Figs. 4 and 5, the range of values observed in the training and testing data sets for these problems is fairly consistent. One of the most prominent observations made from Table 1 is the performance of the neural network based forecasting methods when compared to MA, RW and LR. The MAE of each neural network forecasting method is between 20 and 45 MW for both the training and test data, while the MAE of MA, RW and LR is in the range of 45–105 MW. It is thought that the reason for this large gap in forecast accuracy of the neural network versus non neural network based approaches is due to the nature of the power demand data. The power demand data is very periodic in nature, as illustrated by Fig. 5. The data set contains many steep gradients where the power demand increases and decreases day to day. These rapid increases and decreases in power demand mean that naïve approaches such as MA and RW are unable to make accurate forecasts. These approaches make future predictions purely based on the current value (and historic values for MA) and do not form any model of the problem. These approaches can work well for problems that contain fewer steep gradients such as predicting wind power generated and CO_2 levels, i.e. RW performs third best on the wind and CO_2 prediction problems. Unlike MA and RW, LR does form a model that is used for prediction. As Table 1 shows however, this linear model is inadequate for accurately predicting the power demand data as LR performs worst of all algorithms at forecasting future power demands. Neural networks perform much better on this problem as they can adapt to the steep and frequent gradients in the power demand data and can accurately represent complex non linearities in the data that other methods cannot.

The results presented in the previous section indicate that CMA-ES can train networks to predict multiple time steps into the future with a reasonable accuracy. Fig. 7 shows that the accuracy of the networks diminishes as the networks predict further into future, as would be expected. These graphs show that the accuracy of the network predictions still remains relatively high even when predicting 2.5 h. This is important when scheduling power generators. Power generators can only increase/decrease their power outputs by a limited amount from hour to hour [64]. This task is even more complex when wind power is incorporated to the power generation process due to its stochastic nature [65]. It is vital to know in advance what the power demand and wind power generation will be so that thermal power generators can be scheduled optimally. The accuracy of the CMA-ES neural network forecasting model evaluated in this research can therefore greatly benefit this process by predicting future power demand and wind power generation with a high degree of accuracy.

An interesting side note is that the final peak at the end of the wind power generation test data corresponds to the amount of wind power generated when Storm Ophelia hit Ireland on the 16th of October 2017. As Fig. 4 illustrates, despite the high winds that were experienced in Ireland during this time period, there was a relatively low amount of power generated from wind energy when compared to earlier in the month. This is because wind turbines are not designed to operate in exceedingly strong winds and are shut down to avoid damage. This research utilized a univariate approach whereby predictions were made solely based on the wind power time series data and not meteorological data. Although it was outside of the scope of the research conducted in this paper, the use of meteorological data can enhance the accuracy of wind power generation forecasting models [14]. In the case of the power generated from wind during Storm Ophelia, incorporating wind speed data into the forecast model would have the adverse effect of reducing the accuracy of the model. This is because the high wind speeds would lead the model to believe that more power will be generated from wind energy when in reality there will be less wind energy available as the turbines are not operational. For the majority of the time however, i.e. when there are no extreme events such as storms, incorporating meteorological data such as wind speed will increase the accuracy of the forecast model.

Previous studies have demonstrated how swarm and evolutionary methods such as PSO and DE can be of benefit to forecasting problems. In 2011, Catalao et al. demonstrated that a PSO trained neural network is an effective approach to short term wind power prediction [53]. In 2016, Yang et al. successfully implemented a DE trained neural network for short term power prediction [60]. Özceylan utilized a PSO trained network to predict CO_2 emissions in 2016 [57]. There is also a wealth of literature outlining successful applications of neural networks to forecasting power demand in many countries, e.g. Iran [30]. Neural networks have also successfully been applied to the task of wind power generation prediction [16]. In terms of air pollutants, neural networks have been successfully applied to the task of CO_2 forecasting [66]. There are also examples of forecasting other air pollutants such as nitrogen oxide [37]. The research presented in this paper has a positive impact on each of these previous studies as it shows that CMA-ES is an effective method for evolving neural networks for forecasting. CMA-ES could therefore be applied to neural networks for each of these forecasting problems and give high forecasting accuracy. Corne et al. even suggest using CMA-ES for short term wind forecasting as future work as they suspected that it would provide a high accuracy forecasts [49]. The research presented in this paper confirm that this is the case. The high forecasting accuracy of a CMA-ES can benefit previous studies that have sought to predict wind power generation from local wind farms in Ireland [42].

The neural networks implemented in this research do not consider any other data when making predictions. The predictions are made solely using historic time series data. Many forecast models incorporate other information such as time of day or estimated wind speed [11]. The results presented here would be directly transferable to instances where the network is trained using time series data and other data such as those mentioned. This research demonstrates that CMA-ES is a very effective approach for training neural networks for energy prediction. Studies that utilize data other than time series data to train a neural network for forecasting would also benefit from training the network using the CMA-ES algorithm. CMA-ES converges faster to a network with higher accuracy than PSO, DE and backpropagation. A larger network with more inputs for other data would therefore benefit further from the use of CMA-ES to train its weights as it is the most effective algorithm out of those evaluated here.

8. Conclusion

The primary aim of this research was to investigate if a neural network train with CMA-ES is capable of accurately predicting Ireland's power demand, wind power generation and CO_2 levels. The results obtained indicate that CMA-ES can in fact produce accurate predictions for each of these problems. Moreover CMA-ES performs very competitively when compared to other state of the art approaches. CMA-ES performs significantly better than other evolutionary methods, i.e. PSO and DE, on all problems in terms of accuracy, convergence speed and consistency. The results show that the largest sources of forecast errors come from when there are large changes in the time series data. The neural networks also appear to perform worse when faced with data that it outside of the range that it has trained on, i.e. the CO_2 test data. On this problem, methods such as LR and RW are more robust to these new data points. When predicting power demands, the CMA-ES trained neural network performed significantly better than all other methods on both training and test data. CMA-ES was able to adapt best to the periodic nature of the time series data. When evaluated predicting multiple time steps into the future, the CMA-ES trained network was able to maintain a reasonably high level of accuracy even when predicting 10 time steps (2.5 h) into the future.

In summary, the contributions of this research are:

1. The novel application of CMA-ES to evolve networks to forecast wind power generation, power demand and CO_2 levels.
2. The application of evolutionary neural networks to Ireland's energy sector.
3. CMA-ES significantly outperformed PSO and DE in terms of convergence speed, accuracy and robustness. CMA-ES also outperformed other state of the art approaches on 5 out of 6 data sets.
4. The accuracy of the evolved network's predictions decreases in a mostly linear fashion the further into the future the network attempts to predict. The evolved network does still posses a reasonable level of accuracy when predicting 2.5 h ahead of time.

8.1. Future work

There are many potential routes for future research that have arisen from this research. One such subject of future work would be to evolve networks that take into account wind data when making time series predictions.

Another promising avenue for future research would be to utilize the evolve forecasting model by implementing the model as part of a model predictive controller for scheduling power generators.

Appendix A. Section in appendix

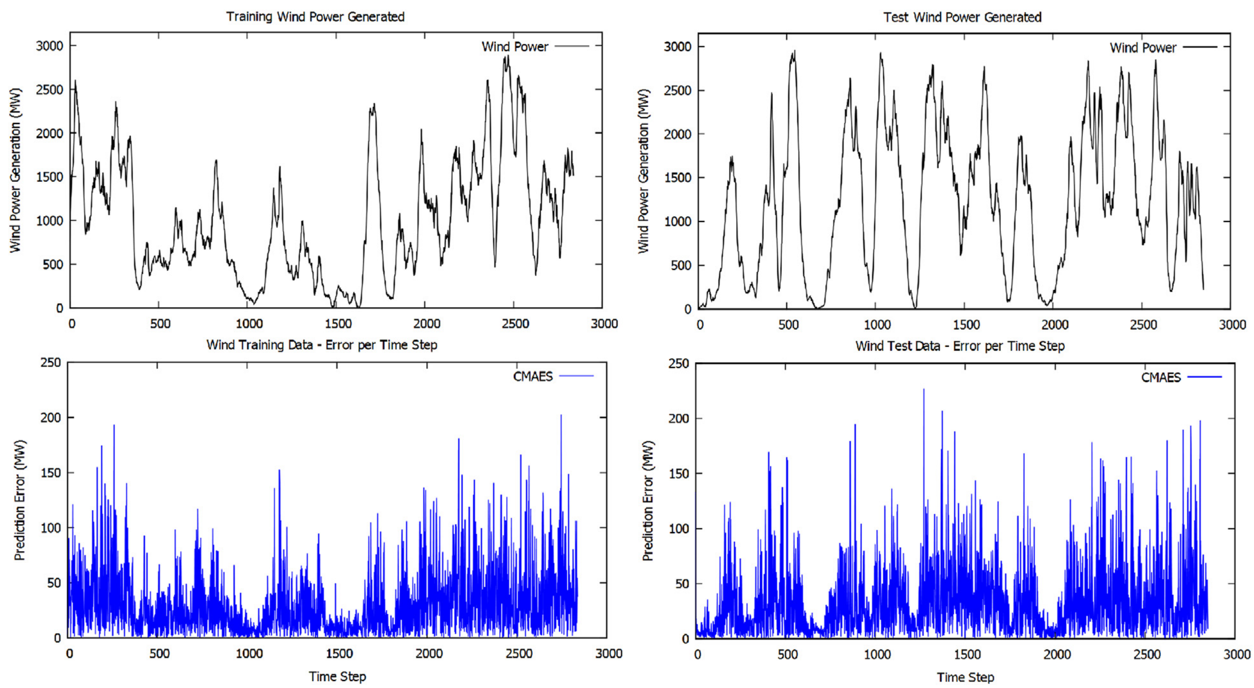

Fig. A.8. Wind Power Generation Error Per Time Step. This figure illustrates the wind power generated along with the error of the Covariance Matrix Adaptation - Evolutionary Strategy trained neural network at each time step.

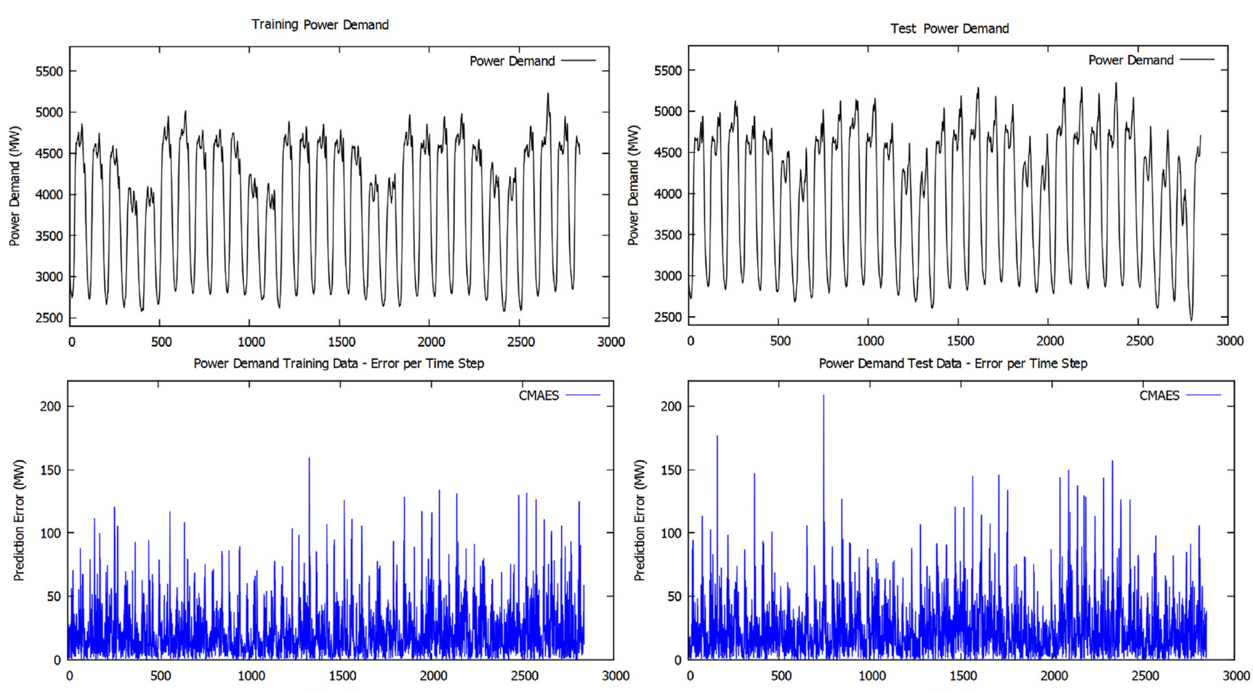

Fig. A.9. Power Demand Error Per Time Step. This figure illustrates the power demand along with the error of the Covariance Matrix Adaptation - Evolutionary Strategy trained neural network at each time step.

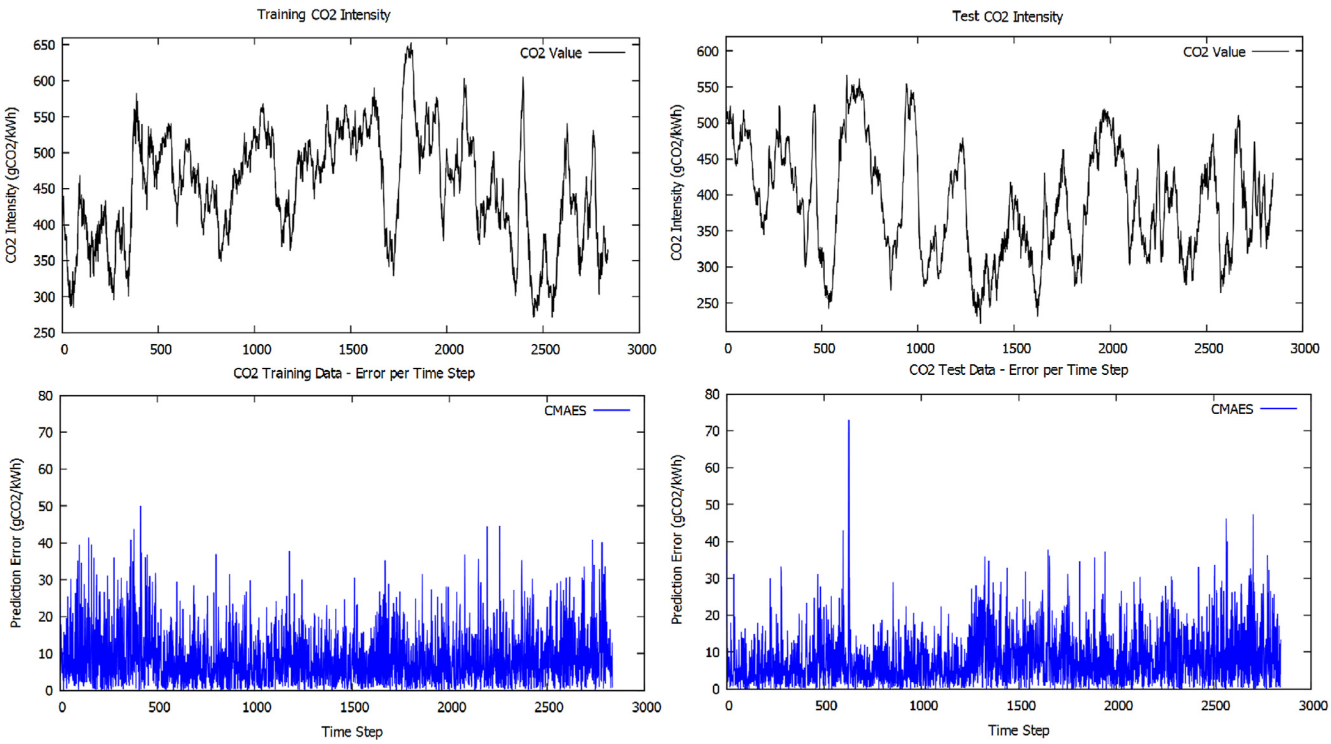

Fig. A.10. Carbon Dioxide Level Error Per Time Step. This figure illustrates the carbon dioxide level along with the error of the Covariance Matrix Adaptation - Evolutionary Strategy trained neural network at each time step.

References

[1] Hansen N, Ostermeier A. Completely derandomized self-adaptation in evolution strategies. Evol Comput 2001;9(2):159–95.

[2] C. S. O. Ireland. Census of population 2016-preliminary results. 2016. http://www.cso.ie/en/releasesandpublications/ep/p-cpr/censusofpopulation2016-preliminaryresults/.

[3] O. N. S. UK. Population estimates for UK. England and Wales, Scotland and Northern Ireland: mid-2016. 2016. https://www.ons.gov.uk/peoplepopulationandcommunity/populationandmigration/populationestimates/bulletins/annualmidyearpopulationestimates/latest.

[4] Corbetta G, Pineda I, Lacal AR, Soede M. IEA wind 2015 annual report. 2016. http://publications.jrc.ec.europa.eu/repository/handle/JRC100976.

[5] Osório G, Lujano-Rojas J, Matias J, Catalão J. A probabilistic approach to solve the economic dispatch problem with intermittent renewable energy sources. Energy 2015;82:949–59.

[6] Boroojeni KG, Amini MH, Bahrami S, Iyengar S, Sarwat AI, Karabasoglu O. A novel multi-time-scale modeling for electric power demand forecasting: from short-term to medium-term horizon. Elec Power Syst Res 2017;142:58–73.

[7] Kaboli SHA, Selvaraj J, Rahim N. Long-term electric energy consumption forecasting via artificial cooperative search algorithm. Energy 2016;115:857–71.

[8] Chen J-F, Wang W-M, Huang C-M. Analysis of an adaptive time-series autoregressive moving-average (arma) model for short-term load forecasting. Elec Power Syst Res 1995;34(3):187–96.

[9] Raza MQ, Baharudin Z, Islam B, Zakariya A, Khir M. Neural network based stlf model to study the seasonal impact of weather and exogenous variables. Res J Appl Sci Eng Technol 2013;6(20):3729–35.

[10] Khashei M, Bijari M, Hejazi SR. Combining seasonal arima models with computational intelligence techniques for time series forecasting. Soft Comput 2012;16(6):1091–105.

[11] Raza MQ, Khosravi A. A review on artificial intelligence based load demand forecasting techniques for smart grid and buildings. Renew Sustain Energy Rev 2015;50:1352–72.

[12] Foley AM, Leahy PG, Marvuglia A, McKeogh EJ. Current methods and advances in forecasting of wind power generation. Renew Energy 2012;37(1):1–8.

[13] Mason K, Duggan J, Howley E. Multi-objective dynamic economic emission dispatch using particle swarm optimisation variants. Neurocomputing 2017;270:188–97.

[14] Al-Yahyai S, Charabi Y, Gastli A. Review of the use of numerical weather prediction (nwp) models for wind energy assessment. Renew Sustain Energy Rev 2010;14(9):3192–8.

[15] Men Z, Yee E, Lien F-S, Wen D, Chen Y. Short-term wind speed and power forecasting using an ensemble of mixture density neural networks. Renew Energy 2016;87:203–11.

[16] Bhaskar K, Singh S. Awnn-assisted wind power forecasting using feed-forward neural network. IEEE Trans Sustainable Energy 2012;3(2):306–15.

[17] Mertz O, Halsnæs K, Olesen JE, Rasmussen K. Adaptation to climate change in developing countries. Environ Manag 2009;43(5):743–52.

[18] Thomas CD, Cameron A, Green RE, Bakkenes M, Beaumont LJ, Collingham YC, Erasmus BF, De Siqueira MF, Grainger A, Hannah L, et al. Extinction risk from climate change. Nature 2004;427(6970):145–8.

[19] Hughes TP, Baird AH, Bellwood DR, Card M, Connolly SR, Folke C, Grosberg R, Hoegh-Guldberg O, Jackson J, Kleypas J, et al. Climate change, human impacts, and the resilience of coral reefs. Science 2003;301(5635):929–33.

[20] S. Levitus, J. I. Antonov, T. P. Boyer, R. A. Locarnini, H. E. Garcia, A. V. Mishonov, Global ocean heat content 1955–2008 in light of recently revealed instrumentation problems, Geophys Res Lett 36(7).

[21] R. Kwok, D. Rothrock, Decline in arctic sea ice thickness from submarine and icesat records: 1958–2008, Geophys Res Lett 36(15).

[22] Bernstein L, Bosch P, Canziani O, Chen Z, Christ R, Davidson O. Climate change 2007: synthesis report. summary for policymakers, in: climate change 2007: synthesis report. Summary for policymakers. IPCC; 2007.

[23] Ganesan P, Rajakarunakaran S, Thirugnanasambandam M, Devaraj D. Artificial neural network model to predict the diesel electric generator performance and exhaust emissions. Energy 2015;83:115–24.

[24] R. Schmalensee, T. M. Stoker, R. A. Judson, World carbon dioxide emissions: 1950–2050, World 80(1).

[25] Wu L, Liu S, Liu D, Fang Z, Xu H. Modelling and forecasting Co_2 emissions in the brics (Brazil, Russia, India, China, and South Africa) countries using a novel multi-variable grey model. Energy 2015;79:489–95.

[26] Bai Y, Li Y, Wang X, Xie J, Li C. Air pollutants concentrations forecasting using back propagation neural network based on wavelet decomposition with meteorological conditions. Atmos Pollut Res 2016;7(3):557–66.

[27] Krzywanski J, Czakiert T, Blaszczuk A, Rajczyk R, Muskala W, Nowak W. A generalized model of SO_2 emissions from large-and small-scale cfb boilers by artificial neural network approach part 2. SO_2 emissions from large-and pilot-scale cfb boilers in O_2/N_2, O_2/CO_2 and O_2/rfg combustion atmospheres. Fuel Process Technol 2015;139:73–85.

[28] Biancofiore F, Verdecchia M, Di Carlo P, Tomassetti B, Aruffo E, Busilacchio M, Bianco S, Di Tommaso S, Colangeli C. Analysis of surface ozone using a recurrent neural network. Sci Total Environ 2015;514:379–87.

[29] Schmidhuber J. Deep learning in neural networks: an overview. Neural Network 2015;61:85–117.

[30] Azadeh A, Ghaderi S, Sohrabkhani S. A simulated-based neural network algorithm for forecasting electrical energy consumption in Iran. Energy Pol 2008;36(7):2637–44.

[31] Ekonomou L. Greek long-term energy consumption prediction using artificial neural networks. Energy 2010;35(2):512–7.
[32] Geem ZW, Roper WE. Energy demand estimation of South Korea using artificial neural network. Energy Pol 2009;37(10):4049–54.
[33] Kermanshahi B. Recurrent neural network for forecasting next 10 years loads of nine Japanese utilities. Neurocomputing 1998;23(1):125–33.
[34] Kialashaki A, Reisel JR. Modeling of the energy demand of the residential sector in the United States using regression models and artificial neural networks. Appl Energy 2013;108:271–80.
[35] Pao H. Forecasting energy consumption in taiwan using hybrid nonlinear models. Energy 2009;34(10):1438–46.
[36] Ye H, Ren Q, Hu X, Lin T, Shi L, Zhang G, Li X. Modeling energy-related CO_2 emissions from office buildings using general regression neural network. Resour Conserv Recycl 2018;129:168–74.
[37] Stamenković LJ, Antanasijević DZ, Ristić M, Perić-Grujić AA, Pocajt VV. Prediction of nitrogen oxides emissions at the national level based on optimized artificial neural network model. Air Qual Atmos Health 2017;10(1):15–23.
[38] S. Qiu, B. Chen, R. Wang, Z. Zhu, Y. Wang, X. Qiu, Atmospheric dispersion prediction and source estimation of hazardous gas using artificial neural network, particle swarm optimization and expectation maximization, Atmos Environ.
[39] Osório G, Matias J, Catalão J. Short-term wind power forecasting using adaptive neuro-fuzzy inference system combined with evolutionary particle swarm optimization, wavelet transform and mutual information. Renew Energy 2015;75:301–7.
[40] Chang G, Lu H, Chang Y, Lee Y. An improved neural network-based approach for short-term wind speed and power forecast. Renew Energy 2017;105:301–11.
[41] Moehrlen C. Uncertainty in wind energy forecasting. Ph.D. thesis. University College Cork; 2004.
[42] Lang S, Möhrlen C, Jørgensen J, Gallachóir B, McKeogh E. Forecasting total wind power generation on the republic of Ireland grid with a multi-scheme ensemble prediction system. Vectors 2006;10:11.
[43] Holland JH. Genetic algorithms and the optimal allocation of trials. SIAM J Comput 1973;2(2):88–105.
[44] Hansen N, Ostermeier A. Adapting arbitrary normal mutation distributions in evolution strategies: the covariance matrix adaptation. In: Evolutionary computation, 1996., proceedings of IEEE international conference on. IEEE; 1996. p. 312–7.
[45] Kämpf JH, Robinson D. A hybrid cma-es and hde optimisation algorithm with application to solar energy potential. Appl Soft Comput 2009;9(2):738–45.
[46] Shir OM, Siedschlag C, Bäck T, Vrakking MJ. Niching in evolution strategies and its application to laser pulse shaping. In: International conference on artificial evolution (evolution artificielle). Springer; 2005. p. 85–96.
[47] Jiang ZH, Werner DH. A compact, wideband circularly polarized co-designed filtering antenna and its application for wearable devices with low sar. IEEE Trans Antenn Propag 2015;63(9):3808–18.
[48] Igel C. Neuroevolution for reinforcement learning using evolution strategies. In: Evolutionary computation, 2003. CEC'03. The 2003 congress on, vol. 4. IEEE; 2003. p. 2588–95.
[49] Corne D, Reynolds A, Galloway S, Owens E, Peacock A. Short term wind speed forecasting with evolved neural networks. In: Proceedings of the 15th annual conference companion on Genetic and evolutionary computation. ACM; 2013. p. 1521–8.
[50] Kennedy J, Eberhart R. Particle swarm optimization. In: Neural networks, 1995. Proceedings., IEEE international conference on, vol. 4; 1995. p. 1942–8. https://doi.org/10.1109/ICNN.1995.488968.
[51] Mason K, Duggan J, Howley E. A meta optimisation analysis of particle swarm optimisation velocity update equations for watershed management learning. Appl Soft Comput 2018;62:148–61.
[52] Welch RL, Ruffing SM, Venayagamoorthy GK. Comparison of feedforward and feedback neural network architectures for short term wind speed prediction. In: Neural networks, 2009. IJCNN 2009. International joint conference on. IEEE; 2009. p. 3335–40.
[53] Catalao J, Pousinho H, Mendes V. Hybrid wavelet-pso-anfis approach for short-term wind power forecasting in Portugal. IEEE Trans Sustain Energy 2011;2(1):50–9.
[54] Bahrami S, Hooshmand R-A, Parastegari M. Short term electric load forecasting by wavelet transform and grey model improved by pso (particle swarm optimization) algorithm. Energy 2014;72:434–42.
[55] Selakov A, Cvijetinović D, Milović L, Mellon S, Bekut D. Hybrid pso–svm method for short-term load forecasting during periods with significant temperature variations in city of burbank. Appl Soft Comput 2014;16:80–8.
[56] Bashir Z, El-Hawary M. Applying wavelets to short-term load forecasting using pso-based neural networks. IEEE Trans Power Syst 2009;24(1):20–7.
[57] Özceylan E. Forecasting CO_2 emission of Turkey: swarm intelligence approaches. Int J Glob Warming 2016;9(3):337–61.
[58] Storn R, Price K. Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. J Global Optim 1997;11(4):341–59.
[59] Meyyappan U, Pandu KDR. Wavelet neural network-based wind-power forecasting in economic dispatch: a differential evolution, bacterial foraging technology, and primal dual-interior point approach. Elec Power Compon Syst 2015;43(13):1467–77.
[60] Yang Y, Chen Y, Wang Y, Li C, Li L. Modelling a combined method based on anfis and neural network improved by de algorithm: a case study for short-term electricity demand forecasting. Appl Soft Comput 2016;49:663–75.
[61] EirGrid. EirGrid energy data all island. 2017. http://www.eirgridgroup.com/.
[62] Kaytez F, Taplamacioglu MC, Cam E, Hardalac F. Forecasting electricity consumption: a comparison of regression analysis, neural networks and least squares support vector machines. Int J Electr Power Energy Syst 2015;67:431–8.
[63] Montgomery DC, Peck EA, Vining GG. Introduction to linear regression analysis, vol. 821. John Wiley & Sons; 2012.
[64] Mason K, Duggan J, Howley E. A multi-objective neural network trained with differential evolution for dynamic economic emission dispatch. Int J Electr Power Energy Syst 2018;100:201–21.
[65] Hetzer J, David CY, Bhattarai K. An economic dispatch model incorporating wind power. IEEE Trans Energy Convers 2008;23(2):603–11.
[66] Pao H-T, Tsai C-M. Modeling and forecasting the co 2 emissions, energy consumption, and economic growth in Brazil. Energy 2011;36(5):2450–8.
[67] Mason K, Duggan J, Howley E. Evolving multi-objective neural networks using differential evolution for dynamic economic emission dispatch. In: Proceedings of the genetic and evolutionary computation conference companion. ACM; 2017. p. 1287–94.